

IO 別冊

MZ-80B活用研究



MZ-80B活用研究

CONTENTS

MZ-80Bを解剖する…………… I/Oエンジンルーム 2

システム/アプリケーション編

モニタ・プログラムの活用法……………COMPAC 30

FORTRAN -MZB……………COMPAC 33

WICS 80B モニタ『TS-1000』…キャリアラボ 佐々木哲哉 47

WICS 80Bインタープリタ…………… キャリアラボ 佐々木哲哉 53

WICS 80B コンパイラ…………… キャリアラボ 山本耕司 61

エディタ・アセンブラ『EDAS』…………… 石橋義徳 73

ラベル付逆アセンブラ……………COMPAC 85

SELF RELOCATABLE DEBUGGER…COMPAC 91

TRACER 80B…HUDSON SOFT 竹部隆司/中本伸一 95

SB-5520の拡張……………山川孝文 101

日本語エラーメッセージ…………… 佐々木雄一 112

COMMAND UP……………ハゲミツ 118

SB-5520ユーティリティ・プログラム…………… 亀谷英行 120

テープ・フォーマットの変換法…………… 大垣泰二 127

PC→MZB CONVERTER…………… FALCON. S 128

MZB→PC CONVERTER…………… 柴田文彦 130

MZ→MZBマシン語CONVERTER

…………… 野口岳郎, I/Oエンジンルーム 132

VOICE SYSTEM 80B……………三屋一彦 134

BASIC VOICE…………… 松井洋介 138

MUSICREATE…………… 柴田文彦 145



グラフィック・キャラクタ・ジェネレータ…………… 柴田文彦 148

PATTERN EDITOR…………… 田中 聡 154

キャラクタ・グラフィック同時コピー…………… 藤原秀樹 157

ブラネタリウム…………… 白木善喜 163

株価・出来高チャート…………… 井上信夫 169

AUTO-2001『CMT-1, CMT-3』

…………… 宇部鉄工マイコンクラブ 梶田 悟 175

AUTO-2001『CMT-2』

…………… 宇部鉄工マイコンクラブ 梶田 悟 193

TV 黒板…………… 高嶋 晃 204

タテ・ヨコ チェック・サム……………46

横8バイトのチェック・サム……………52

ゲーム編

BLOCK くずし…………… ルリタテハ 208

SPACE INVADERS…………… FALCON. S 213

HEAD-ON PART2…………… ルリタテハ 220

グラフィック花札こいこい…………… 高嶋 晃 227

4人ポーカール…………… 大場啓二 240

チェス・プログラム…………… 馬場隆信/COMPAC. 247

BACK GAMMON…………… 大藤謙二 271

CRAZY CLIMBER…………… ばっくまん 280

DONKEY KONG モドキ…………… 居候VALZA 287

DONKEY CLIMBER…………… ばっくまん 297

PUCK MAN…………… ルリタテハ 306

DAN GAME…………… ばっくまん 313



写真1
MZ-80Bシステム

MZ-80Bを 解剖する

I/Oエンジンルーム

MZ-80BはMZ-80K/Cのクリーン・コンピュータの性格を上回る64K RAMを標準実装し、高速かつ強力なBASIC、高分解能グラフィックを搭載しています。ここではそのすばらしい性能を紹介します。

なお、マニュアルには、MZ-80BのIPL, モニタ, 全回路が公開されており、クリーン・コンピュータとしての性格を一層明確にしています。

— 1981年5月号より転載 —

システムの概要

外観

MZ-80BはMZ-80K/Cと同じ構成で、キーボード、CRT(グリーン・ディスプレイ)、カセットが一体となっています(写真1〜3)。

しかし、機能的には大幅な違いがあります。ファンクション・キー、カーソル・キー、テープ・コントロール・キーがキーボード上に配置され、オフィス用として欠かせない「OO」キーを含むテン・キーを持っています(写真4)。

また、通常あまり使用することのない電源SW、IPLリセットSWなどが背面に並んでいます(写真3)。各種インターフェイス・カード(オプション)用の端子もここに窓が開いており、将来システムの拡張時に外観を傷つける

写真2 MZ-80Bの正面

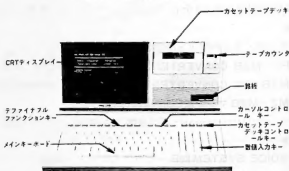
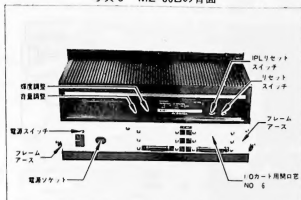


写真3 MZ-80Bの背面



必要はありません。

ハードウェアの構成

詳しいことは次節で述べることにして、ここでは大ざっぱに本体内部を見てもみることにしましょう。

MZ-80BはCPUにZ80A(4MHzタイプ)を使用し、CPUが直接アクセスできるメモリ容量は64KバイトすべてのRAM構成という「クリーン・メモリ・システム」となっています。

システム・ブート用ROM(IPL)2K、キャラクター・RAM(2K)、高性能グラフィックV-RAM(8K:オプション)はメイン・メモリ領域とは切り離されており、必要時にアドレス切り替えによってアクセスできます。

また、CPUへの割り込みは全面的にユーザーに開放されており、PIOのプログラミングによりキーからの割り込み、外部デバイスからの割り込みが受け付けられるようになっているため、高度なプログラミングができるようになっています。

カセット・ディッキは、デジタル・カセットや、オーディオ製品を思わせる電磁メカを使用し、プログラムのセーブ、ロード、データ・ファイルの入出力などをソフトでコントロールできるオートマチック仕様となっています。

また、早送りや望みのプログラムを探す(いわゆるテープの頭出し)APSSを内蔵し、標準オーディオ・テープを使った標準ファイル入出力装置としてデジタル・カセットを上回る機能を持っています。

拡張用I/Oユニットは本体内部にオプションとして組み込むことができます。これによって、本体内部に6枚までインターフェイス・カードを組み込むことができ、システム拡張時に周辺と本体を直接接続できます。

このように、MZ-80Bはクリーン・コンピュータとして、ユーザーが任意にシステムの性格を決定できる数多くの面を持っています。ビジネス、ゲーム用にとユーザーの要求が大きい高分解能グラフィックス、そしてその画面合成などはユーザーのニーズに充分に答えることができるでしょう。

外部への拡張も表1に示すように多数、メーカー・サイドから将来供給されるようです。参考までにその価格を表1に示しますが、従来マイコンの周辺は高いという観点からすると、相当安価となっているのが見受けられます。

これは、システムのグレード・アップを旨とするマイコン・ユーザーを充分に考慮しているものでしょう。

表2-4にMZ-80Bおよび専用DISK、プリンタの仕様を示します。

写真4 MZ-80Bのキーボード部



表1 周辺機器一覧(オプション)

周辺機器名	製品名	標準価格
ドット・プリンタ	MZ-80BP5	¥142,000
ドット・フリンク	MZ-80P4	価格未定
フロッピーディスク	MZ-80BF	¥298,000
マークカートリッジ	MZ-80MCF	¥198,000
14型カラーディスプレイ・ユニット	MZ-80DU	¥294,000
グラフィックRAM I	MZ-8BG	¥39,000
拡張グラフィックRAM II	MZ-8BGK	¥39,000
フロッピー用I/Oカード	MZ-8BF1	¥38,000
拡張I/Oポート	MZ-8BK	¥19,800

表2 MZ-80B仕様

項目	仕様
CPU	SHARP LH0080A(Z80A CPU)
クロック	4MHz(16MHz 4)
メモリ	ROM 2Kバイト(IPL) 2Kバイト(キャラ・ジェネ) RAM 64Kバイト(DRAM)標準実装
CPU周辺	PIO SHARP LH0081A(Z80A PIO) Programmable counter 8253 PPI 8255
ディスプレイ	9インチ グリーン・モニタ キャラクター表示時(8×8ドット・マトリクス) (1,000字(40字×25行)22,000字(80字×25行) グラフィック(オプション) 2ページのグラフィック・エリア ソフト切り替えてグラフィック同上、グラフィックとキャラクター画面の合成可、
カセット・ディッキ	ノーマルタイプのオーディオ・カセットを使用 データ転送速度 1800bps (PWM) ディッキのコントロール 全自動またはマニュアル
音声出力	Max 400mW(440Hz)
キーボード	92キー(ASCIIキーボード、テン・キー、ファンクション・キー、コントロール・キー) ディッキ コントロール・キー
電源	AC100V 50/60Hz
動作環境	気温 0℃-35℃ 湿度 80%以下
寸法・重量	450(W)×520(D)×270(H)mm 16kg

ソフトウェアの構成

MZ-80Bでは64Kバイトのメモリ空間すべてをRAMとしてユーザーに解放しているため、その上で動くソフトウェアはほとんどハードウェアに制約を受けることがありません。これは、MZ-80KやCのような「クリーン・コンピュ

図1 MZ-80Bシステム・ダイアグラム

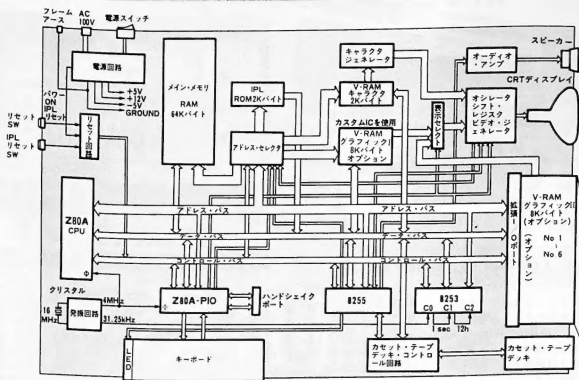


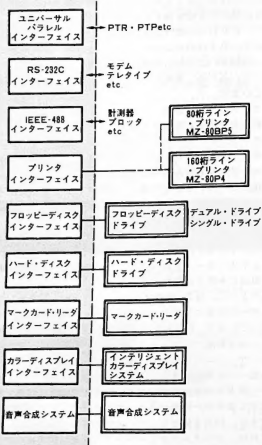
表3 フロッピーディスク MZ-80B仕様

項目	仕様
ドライブ数	2ドライブユニット
ドライブ間の接続	デジタイゼーション方式
記録メディア	5.25インチ両面フレキシブル・ディスク
記録容量	573Kバイト(2ドライブ、両面倍密度)
トラック数	70トラック
セクタ	ソフト・セクター
電源	AC100V 50/60Hz
消費電力	35W
外形寸法	205(W)×320(D)×200(H)mm

表4 ドット・プリンタMZ-80BP5仕様

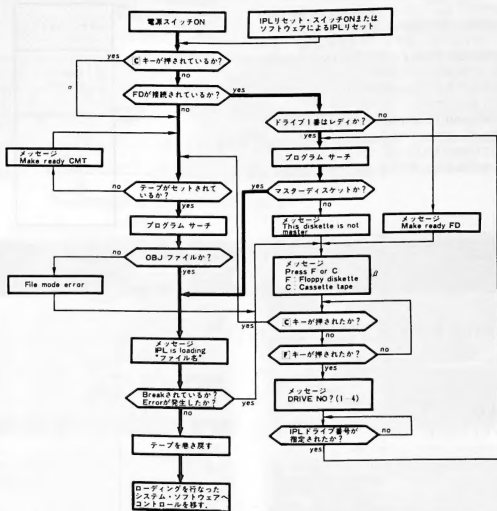
項目	仕様
印字方式	インパクト・ドット・マトリックス
紙送り方式	可変スプロケット・フィード
文字種類	230文字種(アルファベット大文字、小文字、カナ文字他)
文字のドット構成	8(縦)×9(横)ドット・マトリックス(普通サイズ文字ハーフドット方式)
行間隔	1/6インチまたはプログラマブル指定
桁数	80桁、40桁、136桁、68桁、またはプログラマブル
ページ	66行/ページまたはプログラマブル指定
印字速度	80CPS(普通サイズ文字)
印字方向	双方向(ロジカルシーキング付き)
	普通サイズ文字印字(80キャラクタ/行)およびその倍長サイズ文字印字(40キャラクタ/行)
	ビット・イメージ印字およびその他の印字(単方向(左→右))
印字用紙	ファンフォールド紙(4~10インチ幅)・専用カートリッジリボン
インターフェイス	8ビットパラレル・インターフェイス
電源	AC100V 50/60Hz
消費電力	65W
外形寸法	377(W)×318(D)×105(H)mm

ペリフェラル・ファミリー(オプション)



*拡張I/Oポート(本体内部オプション)には、グラフィック・カード2を取り付けた場合、5枚のペリフェラル・カードしか入れることができません。

図2 IPL動作フローチャート



ータ”というシャープが以前から独自に打ち出してきた路線をさらに進めたものといえます。

しかし、このような考え方はシステムに柔軟性をもたらし、少しくとも基本構想を誤ると非常に使いづらいものになってしまうという危険性ははらんでいます。

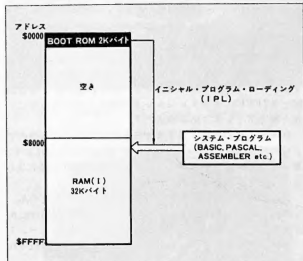
PC-8001やベーシックマスター レベル3ではBASICをROMで持っていて、パワー・オンと同時にBASICが使えるようになっていますが、MZ-80KやCなどではモニターからBASICを読み込まなくてはなりません。

今回のMZ-80Bも、カセットあるいはDISKからBASICを読み込む必要があります。しかし、MZ-80K/Cと違うところは、モニターさえもROMとして持っていないという点です。

MZ-80Bに付いているROMは、キャラジェネを除けばIPL (Initial Program Loader: 2Kバイト) だけです。IPLというのはDISKからシステムをLOADして、制御をそれに移す働きをするもので、ミニコンなどでは古くから使われています。電源投入時のIPLの動作を図2に示しておきます。

IPLはカセットやDISKからモニターやBASICあるいはPASCALなどのシステム・プログラムをLOADし終わると自分自身は、メモリ・マップの上から消え去ってしま

図3 IPL動作時のメモリ・マップ



うので、64Kバイトのアドレス空間はすべてRAMとなります。

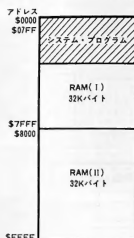
IPLを再起動するためには電源の再投入を行なうか、IPLリセット・スイッチを押さなければなりません。そ

のために、モニタに戻る通常のリセット・スイッチとIPLリセット・スイッチは別々に設けてあります。

IPL起動時のメモリ・マップを図3に示します。図中のBOOT ROMにはIPLが入っていてBASICやPASCAL、アセンブラなどのシステム・プログラムをIPLが読み込むと\$8000番地から32KバイトのRAMが\$0000番地から変わって制御はそちらに移り、メモリ・マップは図4のように変わります。この時点でIPLはRAMの下に隠れてしまうのです。

このようにMZ-80BはRAM領域を最大限に利用できるような設計されているので、本体とDISK、それに将来供給される予定のFDOSやPASCAL、アセンブラなどを組み合わせれば、相当強力なシステムとなることが予想されます。

図4 システム・プログラム ロード後のメモリ・マップ



システムの解説

ハードウェア編

MZ-80Bのハード構成について前に簡単に述べましたが、再度、詳しく見てみることにしましょう。

a. メモリ

メイン・メモリ空間64KバイトはすべてRAMで構成されています。つまり、コンピュータに実行させるプログラムはプログラム・テキストもプログラミング言語(BASIC, PASCALなど)もすべてユーザーの自由な選択により、メイン・メモリ空間に配置できます。

また、ROMによってユーザーが使用できるメモリ領域が狭くなることがないわけです。

b. V-RAM

V-RAMアクセス時には、メイン・メモリとV-RAMの切り替えが行なわれます。この切り替えの方法は2通りあって、図5に示すようにRAMブロック2の後半12Kバイト(\$D000~\$FFFF)をアドレッシング空間から外して、その代わりにV-RAMをアクセス状態とする構成と、図6に示すようにRAMブロック1の後半12Kバイト(\$5000~\$7FFF)をアドレッシング空間から外す代わりにV-RAMをアドレスに結合構成です。

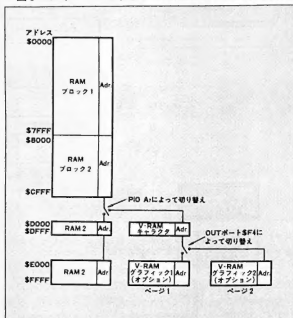
BASICのPRINT文やSET文では主に前者のアドレス切り替えによってデータがV-RAMに送られますが、表プログラムやそのデータがアドレス\$D000以降にある場合は後者が使われるわけです。

このようなアドレスの切り替えはそれぞれPIOのA₀、A₁ポートによって実行され(A₀がアクセスされた場合はA₁は無視される)。さらにグラフィックV-RAMのページ1、ページ2の切り替えは出力ポートSF4によって決められます。

c. キーボード構成

図7はMZ-80Bのコンソールです。メイン・キーボードを中心に右側にテン・キー、上部に10個のユーザー定義可能

図5 メイン・メモリとV-RAMのアドレス切り替え



なファンクション・キーとその機能を書き込んだラベルをセットするラベル。上下左右にカーソルを移動するカーソル・コントロール・キー、カセット・コントロール・キーがグループ別に色分けして配置されています。

メイン・キーボードからはモードの切り替えによってアルファベットの太文字、小文字、太文字の白黒反転文字、数字とその白黒反転文字、カナ文字、シンボル記号、擬似グラフィック・パターンを合計226個のキャラクタをキー入力できます。キーの配列はJIS、ASCIIに準拠し、操作性を良くしています。

d. カセット・デッキ

カセット・テープデッキはIPL操作、プログラム・テキストのセーブ・ロード、データの書き込み、読み出し、巻き戻し、早送りをすべてソフトウェアでコントロールで

図7 キー配置図

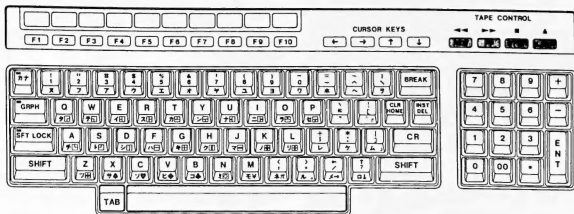


写真5 グラフィック・デモ

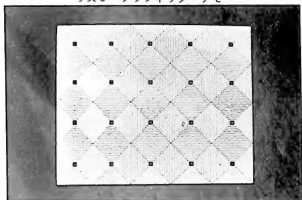


写真6 3次元グラフィックス・デモ

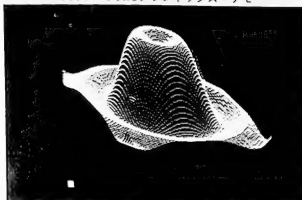


図6 メイン・メモリとV-RAMのアドレス切り替え

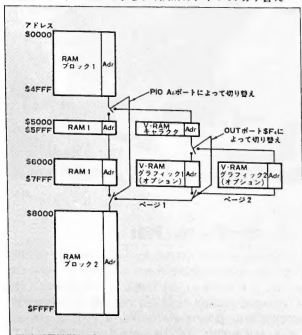


写真7 2つのグラフィックスとキャラクタの合成

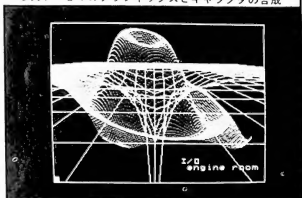
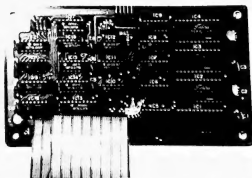


写真8 グラフィック・ポート



きます。

また、コンソール上には巻き戻し、早送り、イジェクトを手動操作するキーが設けられています。

各システム・ソフトのファイル・サーチにはAPSS (Automatic Program Search System)機構があり、早送

図8 V-RAMとCRTディスプレイのつながり

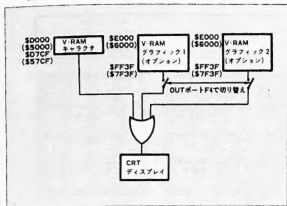


表5 V-RAMの制御

ポート \$F4への 出力データ	V-RAMグラフィック1 INPUT	V-RAMグラフィック1 OUTPUT	V-RAMグラフィック2 INPUT	V-RAMグラフィック2 OUTPUT
0 0	○	×	×	×
0 1	×	×	×	×
0 2	○	○	×	×
0 3	×	○	×	×
0 C	×	×	×	○
0 D	×	×	×	○
0 E	×	○	×	○
0 F	×	○	○	○

注) INPUT ○ V-RAM転送可
 × V-RAM転送不可
 OUTPUT ○ ディスプレイに表示する
 × ディスプレイに表示しない

図9 V-RAMアドレスとCRTディスプレイ



写真9 グラフィック・ボードMZ-8 BGK

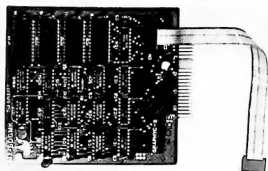
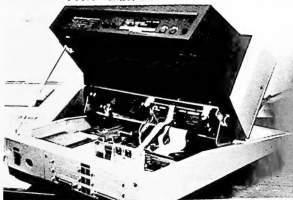


写真10 拡張/OポートMZ-8 BK



ません。

f. オーディオ、時計

カセット・デッキの下のコンソール面にスピーカーを内蔵しており、その音声出力をデータ・エンター・ベル、エラー音、SE（効果音）などに使えます。BASIC、PASCALなどには3オクターブの音域内の音を出せる機能が用意され、音楽の自動演奏もできます。

MZ-80 B 体内には時計機能があり、ソフトウェアによってそのデジタル時計をいつでも呼び出すことができます。

g. オプション・デバイス

MZ-80 Bの拡張デバイスは図1に示してあるとおりです。グラフィック・メモリ・カード1 (MZ-8BG, 写真9) はダイレクトに体内に実装できますが、それ以外のカー

りによって所望のファイルを素早く見つけ出す働きがあります。

カセット・テープ・ファイルによる入出力は2,000bpsと高速になっています。

e. CRTディスプレイ

グリーン・モニタのCRTディスプレイにはキャラクタ表示のほかに、高分解能グラフィック表示を行うことができます。キャラクタの表示は80×25行、または40×25行の2通りの表示が可能です。

グラフィックではオプションのグラフィック・メモリを本体にセットすることにより、320×200ドットの高分解能の表示が可能です。BASIC SB-5520などのサポート・システム・ソフトのグラフィック・コントロール文によりグラフ作成、漢字、仮名表示などが容易にできます。

また、画面全体の反転もソフトウェアによって制御できます。

キャラクタV-RAMとグラフィックV-RAMの関係は図8に示すとおりです。図8のようにV-RAMキャラクタと、グラフィック1または2を同時に表示できます。参考までにV-RAMのアドレスと、CRTディスプレイ上の該当位置との関係を図9に示します。

グラフィック用V-RAMの入出力制御は、OUTポート \$F4 に出力するデータによって表5のように制御することができます。

たとえば、\$F4ポートに03Hを出力して \$E000に01Hを格納したとすると、V-RAMグラフィック2上の \$E000に01Hが転送されますが、ディスプレイ表示は、V-RAMグラフィック1になっているのでディスプレイには表示され

ド(グラフィック・メモリ・カード2、プリンタ・インターフェイスなど)は拡張I/Oポート(MZ-8BK、写真10)を介してやらなければなりません。

拡張I/Oポートを本体内に設けることにより、MZ-80Bは外部に無限に発展させることが可能となります。プリンタやフロッピーディスク、カラーターミナルなどが簡単に接続されるわけです。

なお、この拡張I/Oポートに差すことのできるカードはグラフィック・メモリ・カード2を含め、最高6枚までです。

ソフトウェア編

BASIC(SB-5520, SB-6520)

BASICは本体の発売と同時に、カセット・ベースのSB-5520とDISK BASIC SB-6520(オプション)がモニタSB-1520とともに供給されます。

基本的な機能はMZ-80K/Cに供給されていたSP-5030やSP-6010とほぼ同じですが、スピードはクロックが4MHzのせいもあってか、ベンチマーク・テストの結果もこれまでの他の機種の基本BASICと比べて、非常に高速です。

基本的にはマイクログソフト系のBASICと同じですが画面表示関係の命令などに違いが見られます。表6.7にSB-5520とSB-6520の命令一覧表を示します。

専用プリンタMZ-80BP5に対する画面COPY命令やグラフィック関係の命令も整備されて強力です。

グラフィック命令

MZ-80K/C用のSP-5030やSP-6010とMZ-80B用に今回発表されたSB-5520およびSB-6520との大きな違いは、グラフィックに関する命令が強化されたことでしょう。

MZ-80Bのグラフィック機能はカラーでこそないものの、大変素晴らしいもので、パーソナル・コンピュータにこれだけの機能を持たせることに成功したシャープの努力には頭が下がります。

SB-5520やSB-6520のグラフィック・コントロール機能もこれらの優れたハードウェアを生かすものです。そこで、ここではBASICのグラフィック・コマンドについて、少し説明しておきましょう。

GRAPH命令

書式: GRAPH <la, Ob, C, F>

- a...グラフィック・エリア番号。1または2
- b...グラフィック・エリア番号。1,2,12または0

この命令はグラフィック表示の入出力モードの指定、またはグラフィック・エリアのクリアやセットを行なう命令です。

グラフィック表示の入出力指定というのは、少しばかりわかりにくいかもしれませんが、たとえば、入力モードを1ページに指定したとすると、以下に説明するGRAPH C, GRAPH F, SET, RESET, LINE, BLINK, PATTERNといったような命令は、グラフィック・メモリの第1ページに対して行なわれるようにセットされるということです。

また、グラフィック・メモリの第2ページを出力モードにセットすると、CRT上にグラフィック・メモリの第2ページの内容が表示されます。

表7 MZ-80B DISK BASICの主な追加コマンド

DIR	指定ディスク上のディレクトリ表示
DIR/P	指定ディスク上のディレクトリをプリンタに表示
LOAD	ファイル上のプログラムをロード
LOAD/T	カセット・テープ・ファイルからの読み出し
SAVE	ディスクセットへの書き込み
SAVE/T	カセット・テープ・ファイルへの書き込み
LOCK	ファイルをロックする。
UNLOCK	ファイルのロックを解除する。
RENAME	ファイル名の変更
DELETE	ファイルの削除
CHAIN	ファイル上のプログラムを呼び出しチェインする。
SWAP	ファイル上のプログラムを呼び出しスワップし、終了後もとのテキストにリターンする。
WOPEN≡	シーケンシャル・アクセス・ファイルの書き込みオープン
PRINT≡	シーケンシャル・アクセス・ファイルヘデータ書き込み
CLOSE≡	シーケンシャル・アクセス・ファイルの登録
KILL≡	ファイル・クローズ
ROPEN≡	シーケンシャル・アクセス・ファイルをキャンセルする。
INPUT≡	シーケンシャル・アクセス・ファイルの読み出しオープン
XOPEN≡	ランダム・アクセス・ファイルの書き込み/読み出しオープン
PRINT≡()	ランダム・アクセス・ファイルヘデータ書き込み
INPUT≡()	ランダム・アクセス・ファイルからのデータ読み出し
IF EOF(#)	シーケンシャル・アクセス・ファイルまたはランダム・アクセス・ファイルからのデータ読み込みの際のファイル・エンド検出。
ON ERROR	エラー発生時のジャンプ
IF ERN	エラー番号の判断
IF ERL	エラー発生行番号の判断
RESUME	エラー処理ルーチンからのリターン

いずれにせよ、出力モードがどのような場合でもキャラクター・メモリ(V-RAM)の内容はCRT上に表示され続けているので、グラフィックをONにした場合でも画面にはグラフィックとキャラクターが重なって表示されることとなります。

例 GRAPH I 1...グラフィック・メモリの第1ページを入力モードにする。
GRAPH O 2...グラフィック・メモリの第2ページを出力モードにする。
GRAPH O12...グラフィック・メモリの第1ページおよび第2ページを同時に出力モードにする。
GRAPH O 0...両ページとも出力しない。

また、GRAPH CやGRAPH F命令は(Cは、Clear, FはFillの意味)、それぞれ入力モードに設定されたグラフィック・エリアの全ドットをリセットおよびセットするものです。

SET命令およびRESET命令

書式: SET x, yおよびRESET x, y
x...X座標の数値データ
y...Y座標の数値データ

入力モードに設定されているグラフィック・エリアの任

表6 MZ-80B BASICインタープリタ

直接実行命令	
LOAD	カセット・ファイルのプログラムをロードする。
SAVE	プログラムをカセット・ファイルにセーブする。
VERIFY	テキスト・エリア上のプログラムとカセット・ファイル上のプログラムを比較する。
CLR	変数の内容を0または空にする。
RUN	プログラムを実行する。
LIST	プログラム・リストを表示する。
LIST/P	プログラム・リストをプリンタに出力する。
KLIST	ファンクション・キーのリストを取る。
AUTO	行番号を自動的に発生する。
NEW	プログラムを消去する。
CONT	プログラムの実行を再開する。
MON	コントロールをMONITORに移す。
BOOT	IPLをスタートする。
ステートメント	
GOTO	指定した番号へジャンプする。
GOSUB	指定した行番号のサブルーチンをコールする。
RETURN	サブルーチンの終了
ON GOTO	多岐ジャンプ
ON GOSUB	多岐サブルーチン・コール
IF THEN	条件式の値によって実行する文を変える。
IF GOTO	条件式の値によるジャンプ
IF GOSUB	条件式の値によるサブルーチン・コール
FOR TO STEP	繰り返しループ
NEXT	DATA文のデータを読み込む。
READ	READ文で読み込まれるデータを指し示す。
DATA	DATA文の読み込みポイントを初期化する。
RESTORE	メロディのストリングを鳴らす。
MUSIC	MUSIC文の実行におけるテンポを設定する。
TEMPO	FN関数の定義
DEF FN	ファンクション・キーを設定する。
DEF KEY	グラフィック入出力モードをグラフィック・エリア1か目に指定する。また各エリアのクリア、フィルを指定する。
GRAPH	グラフィック表示でラインを引く。
LINE	グラフィック表示でグラフィック・ラインを引く。
BLINE	グラフィック・エリアのドットをセト
SET	グラフィック・エリアのドットをリセト
RESET	グラフィック・パターンを表示する。
PATTERN	グラフィック表示ポジションを決める。
POSITION	グラフィック表示のポイント情報を得る。
POINT	ポジション・ポイントの水平位置
POSH	ポジション・ポイントの垂直位置
POSV	80キャラ/40キャラ表示、ノーマル/リベース表示、スクローリング・エリアの決定
CONSOLE	アルファベット・キーのソフト・ポジション
CHANGE	大文字/小文字を速にする。
REW	カセットの巻き戻しをする。
FAST	カセットの早送りをする。
WOPEN/T	書き込み用データ・ファイルをオープンする。
ROPEN/T	読み出し用データ・ファイルをオープンする。
CLOSE/T	ファイルをクローズする。
INPUT	キーボードからデータを入力する。
INPUT/T	カセット・ファイルからデータを入力する。
GET	キーボードから現在押されているキーデータを入力する。
PRINT	画面にデータを表示する。
PRINT T	カセット・ファイルにデータを出力する。
PRINT P	プリンタにデータを表示する。
PAGE/P	プリンタの1ページの行数を決める。

IMAGE/P	プリンタにイメージ・プリントを行なう。
COPY/P	ディスプレイ表示のコピーを取る。
SIZE	未使用のテキスト・エリアのサイズを示す。
TI S	内蔵時計のストリング変数
CURSOR	カーソル移動(2次元)
CSRH	カーソルの水平位置
CSRV	カーソルの垂直位置
TAB	カーソル移動(タブレーション・1次元)
(PRINT)SPACES	スペースの表示
DIM	配列変数のディメンジョン定義
STOP	プログラムの実行を停止する。
END	プログラムの実行を終了する。
REM	コメントを書く。
LET	代入式を表す。
INP	I/Oポートからデータの読み込み。
OUT	I/Oポートへのデータの書き出し。

機械種とのリンク命令	
LIMIT	BASICプログラム・エリアの制限
PEEK	メモリ内データを直接読み出す。
POKE	メモリ内に直接データを書き込む。
USR	メモリ内の指定番地を直接コールする。

ストリング処理命令	
LEFT S	ストリングの左のn文字
RIGHT S	ストリングの右のn文字
MID S	ストリングの中のn文字目からm文字
LEN	ストリングの文字数
ASC	ストリングをASCIIコード(10進)に変換
CHR S	ASCIIコードをストリングに変換
VAL	数字ストリングを数値に変換
STR S	数値を数字ストリングに変換
STRING S	連続した文字を得る。
CHARACTER S	キャラクタ表示上の情報を得る。

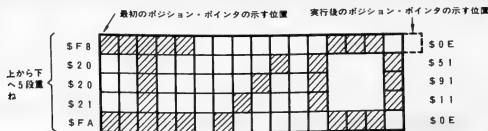
ファンクション	
ABS	絶対値
SGN	符号
INT	切り捨てによる整数化
SIN	サイン
COS	コサイン
TAN	タンジェント
ATN	アークタンジェント
SQR	スクエアルート
EXP	指数関数
LOG	常用対数
LN	自然対数
RND	乱数

その他の仕様・定数など	
行番号	1-65535
ファイル・ネーム	最大16文字
CURSORコマンドにおけるXとY	X=0-79, Y=0-24
SET-RESETにおけるXとY	X=0-319, Y=0-199
FOR NEXTのネスティング	最大16
GOSUBにおけるネスティング	最大16
DEF FNにおけるネスティング	最大6
ストリング比較: ASCIIコードの値による比較	
スクローリング: スクロール範囲を決めることが可能	
文(1つの行番号を持つステートメントの号)の長さ	最大160桁

意の1ドットをセト、リセトする命令で座標は左上隅が(0, 0)、右下隅が(319, 199)です。0 ≤ x ≤ 16,383, 0 ≤ y ≤ 16,383の範囲であれば、エラーにはなりません。

LINE命令およびBLINE命令

書式	LINE	x ₁ , y ₁ , x ₂ , y ₂ , <x ₃ , y ₃ ... x _n , y _n >
	BLINE	x ₁ , y ₁ , x ₂ , y ₂ , <x ₃ , y ₃ ... x _n , y _n >
		x ₁ , ... X座標の数値データ
		y ₁ , ... Y座標の数値データ



上のようなパターンのストリングス・データは、

$\text{CHR} \$ (\$ F 8) + \text{CHR} \$ (\$ 2 0) + \text{CHR} \$ (\$ 2 0) + \text{CHR} \$ (\$ 2 1) + \text{CHR} \$ (\$ F A) + \text{CHR} \$ (\$ 0 E) + \text{CHR} \$ (\$ 5 1) + \text{CHR} \$ (\$ 9 1) + \text{CHR} \$ (\$ 1 1) + \text{CHR} \$ (\$ 0 E)$

となります。これが A \$ というストリングスに代入されていたとすれば

上から下へ5段重ね
PATTERN - 5, A \$

を実行すると、ポジション・ポイントの位置に上のパターンが表示されます。

例 1

Toho Yohogi Kougakusha SD engine room

入力モードに設定されているグラフィック・エリアに任意のラインを引く命令で、 x_1 、 y_1 が組となって最初の組の示す座標点から順々に線を引いていきます。

BLINE 命令は黒い線、リセットされた線分を引きます。

POSITION 命令

書式: POSITION x , y

x X座標の数値データ

y Y座標の数値データ

PATTERN文によるドット・パターンの表示位置を決めるために、ポジション・ポイントの位置を設定します。

PATTERN 命令

書式: PATTERN (x_1 , x_2 \$ (x_3 , x_4 \$) ... (x_n , x_n \$)

x_1 数値データ: ドット・パターンの積み重ね段数。

x_2 \$ ストリングス・データ: 8ビット単位のパターンを表わすデータ並び。

入力モードに設定されているグラフィック・エリアに任意のグラフィック・パターンを書きます。例1はPATTERN命令を使ったものです。グラフィック・パターンはポジション・ポイントの示す位置を起点として、積み重ね段数を x_1 、ドット・パターンのデータ並びを x_2 \$ の内容として描かれます。

描くべきドット・パターンは8ビット単位で与えられ、各ドット・パターンはポジション・ポイントの示す位置からX軸の正の方向に順に、ドットのセット、リセットを決めます(例2)。

ドット・パターンの積み重ねについては少しばかり複雑で、まず積み重ねる方向を決めます。下から上へ積み重ねる

例 2



このようなドットパターンを定義するには、セットするべきビットを1、リセットするビットを0として8ドットを2進数で表わし、さらにそれを16進数または10進数に変換した後、ストリングス・データ化します。

例えば、上のパターンを16進数で表わせば5 Bですからそれをストリングス化したCHR \$(5 B)がドットパターン・データになります。

るには x_1 を正の値とし、逆に上から下に積み重ねるには負の値とします。

積み重ねは上向きあるいは下向きに $|x_1|$ 段実行した後、ポジション・ポイントがX軸の正の方向へ8ドット分移り、先のドット表示のすぐ右隣りに続けて描かれます。実際にパターン・データをどのように定義するのは、例3をご覧ください。

POINT 命令

書式: POINT (x_1 , y_1)

x 数値データ: X座標

y 数値データ: Y座標

x 、 y で指定されるグラフィック・エリアの任意の座標点がセットされているか、リセットされているかの情報を与える関数です。与えられた情報は次のような意味を持ちます。

0...グラフィック・エリアのページ1、2ともリセットされている。

1...グラフィック・エリアのページ1のみセットされている。

2...グラフィック・エリアのページ2のみセットされている。

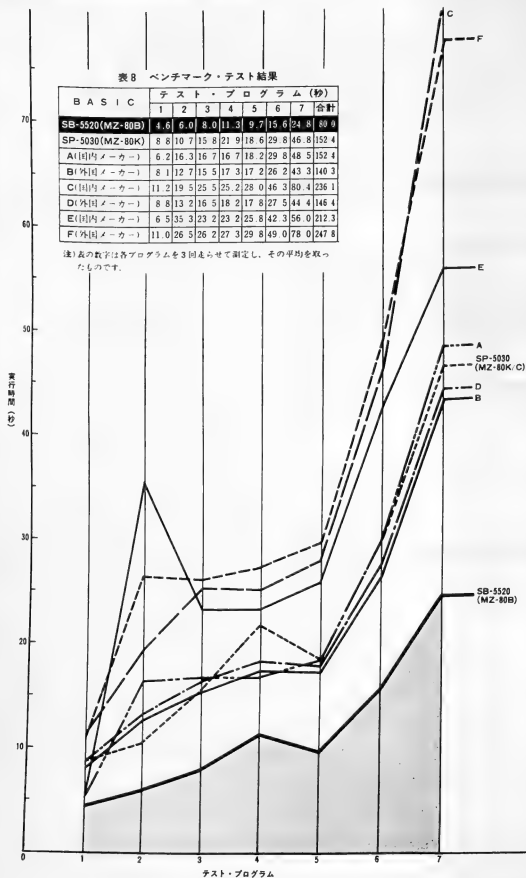
3...グラフィック・エリアのページ1、2ともセットされている。

図10 ベンチマーク・テスト結果

表8 ベンチマーク・テスト結果

BASIC	テスト・プログラム (秒)							合計
	1	2	3	4	5	6	7	
SB-5520(MZ-80B)	4.6	6.0	8.0	11.3	9.7	15.6	24.8	89.0
SP-5030(MZ-80K)	8.8	10.7	15.8	21.9	18.6	29.8	46.8	152.4
A(国内メーカー)	6.2	16.3	16.7	16.7	18.2	29.8	48.5	152.4
B(外国メーカー)	8.1	12.7	15.5	17.3	17.2	26.2	43.3	140.3
C(国内メーカー)	11.2	19.5	25.5	25.2	28.0	46.3	80.4	236.1
D(外国メーカー)	8.8	13.2	16.5	18.2	17.8	27.5	44.4	146.4
E(国内メーカー)	6.5	35.3	23.2	23.2	25.8	42.3	56.0	212.3
F(外国メーカー)	11.0	26.5	26.2	27.3	29.8	49.0	78.0	247.8

注) 表の数字は各プログラムを3回走らせて測定し、その平均を取ったものです。



TEST 1 10 PRINT "START" 20 I=0 30 I=I+1 40 IF I<1000 THEN 30 50 PRINT "END" 60 END	TEST 4 10 PRINT "START" 20 I=0 30 I=I+1 40 A=I/2*3+4-5 50 IF I<1000 THEN 30 60 PRINT "END" 70 END	TEST 6 10 PRINT "START" 20 DIM A(5) 30 I=0 40 I=I+1 50 B=I/I*I+I-1 60 GOSUB 120 70 FOR N=1 TO 5 80 NEXT N 90 IF I<1000 THEN 40 100 PRINT "END" 110 END 120 RETURN
TEST 2 10 PRINT "START" 20 FOR I=1 TO 10000 30 NEXT I 40 PRINT "END" 50 END	TEST 5 10 PRINT "START" 20 I=0 30 I=I+1 40 B=I/I*I+I-1 50 GOSUB 90 60 IF I<1000 THEN 30 70 PRINT "END" 80 END 90 RETURN	TEST 7 10 PRINT "START" 20 DIM A(5) 30 I=0 40 I=I+1 50 B=I/I*I+I-1 60 GOSUB 130 70 FOR N=1 TO 5 80 NEXT N 90 A(N)=N 100 IF I<1000 THEN 40 110 PRINT "END" 120 END 130 RETURN
TEST 3 10 PRINT "START" 20 I=0 30 I=I+1 40 A=I/I*I+I-1 50 IF I<1000 THEN 30 60 PRINT "END" 70 END		

システム変数POSHおよびPOSV

POSHはグラフィック・エリアIのホジション・ポイントのX座標をりえるシステム変数で、取り得る値は0～319です。同様にPOSVはY座標の値をりえ、取り得る値は0～199です。

ベンチマーク・テスト

前にも書きましたが、MZ-80B用のBASIC、SB 5520およびDISK BASIC SB 6520は人変高速で、ベンチマーク・テストの結果も表8および図10のようになりました。スピードだけでBASICを評価することはできませんが、やはり速いにこしたことはありません。なお、今回の評価用に使ったベンチマーク・テスト プログラムをリスト1に示します。

モニタ

MZ 80K/Cのモニタは、基本的にはプログラムのLO

表9 モニタ・コマンド表

コマンド名	内 容
M(MEMORY CORECTION)	メモリ内容の変更
D(MEMORY DUMP)	メモリ内容の表示
J(JUMP)	任意のアドレスに制御を移す
S(SAVE)	メモリ内容をセーブ・テープ・SAVE
V(VERIFY)	メモリ内容とテープのVERIFY
L(LOAD)	メモリテープからLOAD

ADとマシン語のプログラムの実行、それにDISKのブートの機能しか持ち合わせていませんでした。しかし、MZ-80Bのモニタ(SB-1520)では、ごく一般的マシン語のダンプ、チェンジなどの機能を持ったものになり、メモリ内容の変更なども、いちいちマシン・ランゲージなどを読み込まなくても済むようになりました。モニタの主な機能を表9に示します。

しかし、何よりも心強いのは、モニタの内容をソース・リストの形で公開していることです。IPLもやはりソース・リストを公開しているので、回路図の公開とともに、これらMZ-80Bに対するシャープの自信の表われかもしれません。

MZ-80K/Cとの互換性

MZ-80K/C→B変換プログラムの使用例

MZ 80BはMZ 80K Cとモニタ・サブルーチンやBASICの文法がほとんど同じなので、ソフトに互換性があります(表10)。

しかし、カセットのボーレートの違いなどにより、そのまま読み込んで走らせることはできません。

いまでも開発されてきた膨大な量のMZ 80K Cのソフトウェア(BASIC・カセット)をそのまま読み込み、MZ 80B用にしてしまうソフトがCONVERTER K・C→B(オプション)です。

使用方法

まず、ディスクまたはカセットからCONVERTER K・C→Bをロードし、走らせます。次にK C用のソフト(ここではコムパックのMZ用平仮名エイリアン)をロードすれば自動的に変換します。

表11 ASCIIコード表

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NULL			0	@	P		P			=	V	Y	タ	ミ	Z	O
1		↓	!	!	A	Q	a	q		↓	¥	。	ア	チ	ム	A	I
2		↑	"	2	B	R	b	r		↑	U		イ	ツ	メ	B	2
3		→	#	3	C	S	c	s		→	●		ウ	テ	モ	C	3
4		←	\$	4	D	T	d	t		←	○	W	エ	ト	ヤ	D	4
5	NAME		%	5	E	U	e	u		♠	□	X	オ	ナ	ユ	E	5
6	CLR		&	6	F	V	f	v		♥	□	ラ	カ	ニ	ヨ	F	6
7	DEL		'	7	G	W	g	w		♦	□	フ	キ	ヌ	ラ	G	7
8	INS1		(8	H	X	h	x		♣	□	イ	ク	ネ	リ	H	8
9	GNP)	9	I	Y	i	y		♠	+	ウ	ケ	ノ	ル	I	9
A	ST		*	:	J	Z	j	z		♠		エ	コ	ハ	レ	J	P
B			+	;	K	[k	[♠	—	オ	サ	ヒ	ロ	K	Q
C	カナ		,	<	L	\	l			♠	—	ヤ	シ	フ	ワ	L	R
D			-	=	M]	m]		♠	—	ユ	ス	ヘ	ソ	M	S
E	EXP		.	>	N	^	n	~		♠	—	ヨ	セ	ホ		N	T
F	RT DEL		/	?	O	_	o	↓		♠	—	ツ	ソ	マ	°	O	π

ト
4
ビ
ト

表10 モニタ・サブルーチン比較

サブルーチン名	MZ-80B アドレス	MZ-80K/C アドレス	機 能
CALL LETNL	\$0A2E	\$0006	行を変え、次の行にカーソルを セット
CALL PRNTS	\$08C4	\$000C	スペース1個表示
CALL PRNT	\$08C6	\$0012	AレジスタのASCIIを表示、 両面コントロール可。
CALL MSG	\$0889	\$0015	(DE)以降のASCIIを0F まで表示、コントロール可。
CALL BELL	\$0F14	\$008E	エントリレベルを鳴らす(ラの 音、約440Hz)
CALL MELDY	\$0F3F	\$0030	MUSIC処理
CALL XTEMP	\$0E50	\$0041	テンポ、Aレジスタの値により 変化(01-07)。
CALL SOUT	\$0F22	なし	BCに音長、Hに音型をセッ トし、コールすれば音を発行
CALL TIMST	\$0E5E	\$0033	タイマをセット
CALL TIMRD	\$0EA9	\$003B	DEにタイマ時間をセット(秒)
CALL BRKEY	\$0562	\$001E	シフト・ブレイク検出
CALL GETL	\$06A4	\$0003	キーボードより1行入力
CALL GETKY	\$0832	\$001B	キーボード・スキャンして入力 をASCIIにして→A
CALL ASC	\$05F3	\$03DA	Aレジスタの下位4ビットを16 進とみなしASCIIに変換し→A
CALL HEX	\$05FD	\$03F9	Aの下位4ビットを16進にして→A
CALL HLHEX	\$0614	\$0410	(DE)以降16進4桁(ASCII) を2進にして→HL
CALL 2HEX	\$0623	\$041F	(DE)以降16進2桁(ASCII) を2進にして→A
CALL *PONT	\$0C29	\$0FB1	カーソル位置をHLにセット

変換されたリストを見ると、ステートメントはもちろんのこと、リストにはMZ 80Bのキャラクターコードのキャ

写真11 平安京エイリアン・ゲームの画面(その1)

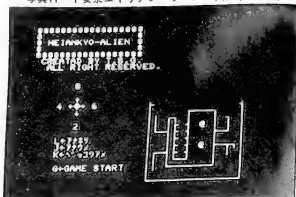
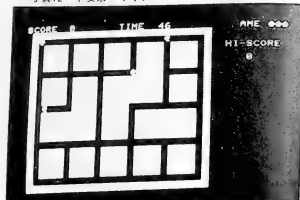


写真12 平安京エイリアン・ゲームの画面(その2)



ラクタが出ています。MZ 80Bでは、直接キーインできな

いカーソル・コントロール・マーク(ⒺⒻⒼⒽ)も表わされます。もっと感心したのは、マシン語とリンクするときのPEEK文、POKE文、USR文は数値を未定義にしようというのです。いくらMZ同士とはいえ、モニタの内容は少し違います。

ゲーム中のエイリアン、機非道使、穴などはV-RAMに直接アクセスしているのです。

POKE %%%%%%%%%, %%

と直されます。幸いにもV-RAMの番地はK/Cと同じく、\$D000番地から入っているのて簡単に直せますが、MZ-80Bとキャラクタが違う(表11)ので、

エイリアン: Ⓔ→Ⓖ

機非道使: Ⓔ→Ⓗ

に直しました。

★V-RAMについて

MZ-80Bでは、ユーザーRAMエリアが64Kフルに使用できるようになっているために、V-RAMは自動的にPIOによって切り替えられるようになっています。そこで、V-RAMとの切り替えを行なっているPIOのポート\$E8(232)のAをOUT文を使って、"H"にします。すると\$D000からのRAMがV-RAMに代わり、PEEK、POKEを行なえるようになります(リスト2)。

このときに注意しなければならない点は、PRINT文によって、画面上へディスプレイするときには必ずOUT文によって元の状態に戻しておかなければなりません。もし戻さないときは、PRINT文のところでリセット(モニタSB-1520へ飛ぶ)してしまいます。

また、BREAKをかけたときも同じです。

★キャラクタについて

K'にあってBにないキャラクタがそのままコンバートされると、違うキャラクタに化けることがあります(これはキャラクタの違いによります)。これを適当なキャラクタに手作業で1つずつ書き直してください(平安京エイリアンは一部DEMOを削除しました)。

カーソル・コントロール・マークが残っているのは有難いことです。しかし、直接のキーインはできません。もちろん[C]はクリア、[H]はホーム、←、→、↑、↓はカーソルの移動方向です。

RUN

操作しやすくするために機非道使を動かすのはテン・キーを使いました。

Ⓔ-上, Ⓖ-左, Ⓖ-右, Ⓔ-下

穴掘り、穴埋め、べっこうあめは従来の通りです。

Ⓔ-穴ほり, Ⓔ-べっこうあめ, Ⓔ-穴うめ

右手と左手が逆になりますが、かえてこの方がゲームはしやすいと思います。

キーを押し続けなくてもその方向を維持して進みます。ゲームの途中でBREAKをかけると、OUT命令の関係でモニタ(SB-1520)へ飛ぶことがあります。

そのときはⒺコマンドで\$1300へ飛ばせば、BASICがホット・スタートします。しかし、ワーク・エリアがイニシャライズされてしまいます。

そのためにファンクション・キーが未定義になってしまいうのです(ファンクション・キーの設定はコールド・スタ

リスト2 POKE文でV-RAMをコントロールする例

```
10 OUT$232,128
20 POKE$D000,30
30 OUT$232,0
40 PRINT"ABC"
50 END
```

写真13 ブラック・ホールの画面(その1)

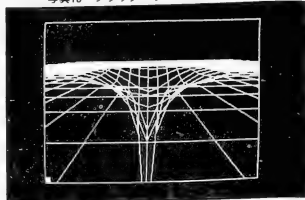


写真14 ブラック・ホールの画面(その2)

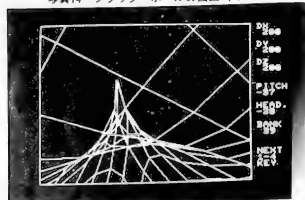


表12 文法上の変更点

ベーシックマスター L3	MZ-80B
CLS	PRINT CHR\$(6)
SPC(n)	SPACE\$(n)
LOCATE X,Y	CURSOR X,Y
DIM(n)	DIM(INT(n/255), n INT(n/255)*255)

ートのときにイニシャライズと同時に進められる)。したがって、ファンクションの再設定をする必要があります。

HIRESグラフィックの使用例

MZ-80BはオプションのグラフィックV-RAMボードを増設することにより、320×200ドットのグラフィックが可能です。このグラフィック機能の使用例として横山英二氏のL3用3次元グラフィック(1/O誌'81年4月号)を移植してみました。


```

3020 TEMP06
3030 MUSIC" 67R264R164R168R2A66ASR26Y
3100 IFS:HSTHENRS=S:OUT232,0:PRINT"#####";S
3106 OUT232,0:PRINT"B"
3110 FORI=0T020
3120 PRINT" ":SPACE(26)
3130 NEXTI
3140 OUT232,128:FORI=1TON:POKEI(1),64:NEXT
3150 OUT232,0:PRINT"#####YOUR GAME IS OVER"
3160 FORI=1T05
3165 GOSUB4100
3170 PRINT" #";TAB(19);" "
3180 GOSUB4100
3190 PRINT" #";TAB(19);"OVER"
3200 NEXTI
3210 PRINT"#####PLAY AGAIN?"
3215 I=0
3230 GETB#
3240 IFB#="Y" THEN70
3250 IFB#="N" THEN3900
3260 I=I+1:IFI(200) THEN3230
3270 GOT065
3900 PRINT"#####GOOD BYE"
3910 END
4000 REM WAIT
4010 FORJ=1T05:NEXTJ:RETURN
4100 REM WAIT
4110 FORJ=1T0300:NEXTJ:RETURN
5000 REM 777
5010 OUT232,0:PRINT" #";
5220 PRINT"#####";
5230 PRINT" # "
5240 PRINT" # HEIANKYO-ALIEN # "
5250 PRINT" # "
5260 PRINT"#####";
5270 PRINT" CREATAD BY T.S.G."
5280 PRINT" ALL RIGHT RESERVED."
5290 PRINTTAB(7);"##### ( # ) #####";
5300 PRINT"#####777";
5310 PRINT"#####777";
5320 PRINT"#####777";
5330 PRINT"#####";
5340 PRINT" # "
5360 PRINTTAB(19);
5370 FORI=1T010
5380 PRINT" | #####";
5390 NEXTI:PRINT" # ";
5400 PRINT" #";TAB(20);
5410 FORI=1T06
5420 PRINT" #####";
5430 NEXTI
5440 PRINT" #####";
5450 PRINT
5460 PRINTTAB(20);" # "
5470 PRINTTAB(20);" # "
5480 PRINTTAB(20);" # "

```

```

5490 PRINTTAB(20);" # "
5500 PRINTTAB(20);" # "
5510 PRINTTAB(20);" # "
5520 PRINTTAB(20);" # "
5530 PRINTTAB(20);" # "
5540 PRINTTAB(20);" # "
5550 PRINT"#####GAME START"
5570 GETB#;IFB#="G" THENRETURN
5600 GOT05570
6000 REM LOAD
6010 OUT232,0:PRINT"SCORE #";S
6015 PRINT" #";TAB(15);"TIME "
6020 PRINT" # "
6025 FORI=1T04
6030 PRINT" "
6040 PRINT" "
6050 PRINT" "
6060 PRINT" "
6070 PRINT" "
6080 NEXTI
6090 PRINT" "
6100 PRINT" #:OUT232,128
6110 FORJ=1T03
6120 I=S3337+INT(RND(1)*3)*200+INT(RND(1)*4)*85
6130 IFPEEK(1) < 32 THEN6120
6140 IFPEEK(1-5)+PEEK(1-5)+PEEK(1+200)+PEEK(1-200) < 128 THEN6120
6150 POKEI,30:POKEI+1,30:POKEI+2,30:POKEI-1,30
6160 NEXT J
6170 FORJ=1T03
6180 I=S3415+INT(RND(1)*4)*200+INT(RND(1)*4)*85
6190 IFPEEK(1) < 32 THEN6180
6200 IFPEEK(1-5)+PEEK(1-5)+PEEK(1+200)+PEEK(1-200) < 128 THEN6180
6210 POKEI,40:POKEI+40,30:POKEI+80,30:POKEI-40,30
6220 NEXT J
6230 I=INT(RND(1)*2)+2:O=E(1):H=H(1)
6240 RETURN
6500 REM INITI-1
6510 DIMX(8),D(8),C(8),E(3),H(3)
6520 H(3)=136:H(2)=136:H(1)=136:H(1)=136
6530 E(0)=1:E(1)=40:E(2)=1:E(3)=-40
6540 HS=0
6550 RETURN
6600 REM INITI-2
6610 GOSUB6000:D(1)=INT(RND(1)*2):IFW=1 THEN6660
6620 FORI=2TON:D(1)=1-D(1-1):NEXTI
6660 FORI=1TON:X(1)=S3330:NEXTI
6670 POKE5330,64:FORI=1TON:GOSUB50:NEXTI
6680 M=S4155:POKEH,H
6900 MUSIC"CAES+C
6910 RETURN

```

MZ-80Bの将来

これまで解説してきたようにMZ-80Bはいままでのパソコンを超える面をいくつか持っています。

たとえば、フロッピーディスクは倍密度に、カセットは高速でオートマッチングになっている点です。また、信頼性追求のため、フラット・ケーブルからシールド・ケーブルにしています。

ここでも買かれたクリーン構造はこれからのパソコンのひとつの方向を示しているようです。

64Kバイトフル・メモリはソフトの発展性にはかせな

いものでしょう。将来、FORTRANなどのコンパイラが走ることは充分考えられます。

しかし、V-RAMを64Kバイト以外に配置しているので、PEEK、POKE文で画面コントロールするとき、いちいちOUT命令を使わなくてはならない点は改善の余地がありそうです。

たとえば、OUT命令+PEEK(POKE)命令はOUT命令を自動的にこなうVPEEK、VPOKEなどのコマンドの追加は、BASICインタプリタがRAM上に

あるため簡単に行なえるでしょう。

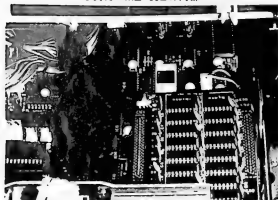
キャラクタV-RAMとグラフィックV-RAMを合成して出力できることはすばらしいアイデアです。

たとえば、グラフィックで線を描かせ、キャラクタV-RAMでワクを気にせず作表したりキャラクタV-RAMでX軸、Y軸を描き、グラフィックV-RAMでグラフを作制することや、2つのグラフィックを使ってCADの利用なども考えられます。

このように、MZ-80Bはビジネスにも充分使いこなせるよう考えられています。また、割り込みはすべてユーザーに開放されているので、CPUの機能を最大限に利用した高度な応用も考えられます。

これだけの機能が¥278,000で備えていては、もうハードを自作する時代は終わってしまったようです。MZ-80Bはビジネスにホビーにと息の長いパソコンになるでしょう。

写真15 MZ-80Bの内部



例14 MB-80Bによる漢字のプリント・アウト例

百人一首

秋の田のかりほの庵のともをあらみわがころもではつゆにぬれつつ
春すぎて夏ににけらし白妙のころもほすてふあまのかぐ山
足引の山鳥の風のしだりおのながながしよとわたりかきねん
田子の浦にうち出てみれば白妙のふじのたかねに雪はふりつつ
おくやまにもよじ露分なく庵の声きくときぞあきは悲しき
かささぎのわたせる橋にくしもしろきをみればよぞふけにける

百人一首

秋の田のかりほの庵のともをあらみわがころもではつゆにぬれつつ
春すぎて夏ににけらし白妙のころもほすてふあまのかぐ山
足引の山鳥の風のしだりおのながながしよとわたりかきねん
田子の浦にうち出てみれば白妙のふじのたかねに雪はふりつつ
おくやまにもよじ露分なく庵の声きくときぞあきは悲しき
かささぎのわたせる橋にくしもしろきをみればよぞふけにける

MZ-80Bの全回路図

■MZ-80B本体の回路図

- (1) CPUボード・ブロック1: CPU信号系..... 19
- (2) CPUボード・ブロック2..... 20
- (3) CPUボード・ブロック3: 8255, PIO信号系..... 21
- (4) CPUボード・ブロック4: RAMブロック..... 22
- (5) CPUボード・ブロック5..... 23
- (6) CRTディスプレイコントロール部..... 24
- (7) カセットテープデッキコントロール部..... 25
- (8) 電源回路..... 26

■グラフィックメモ리카ード1, MZ-8BG(オプション)の回路図..... 27

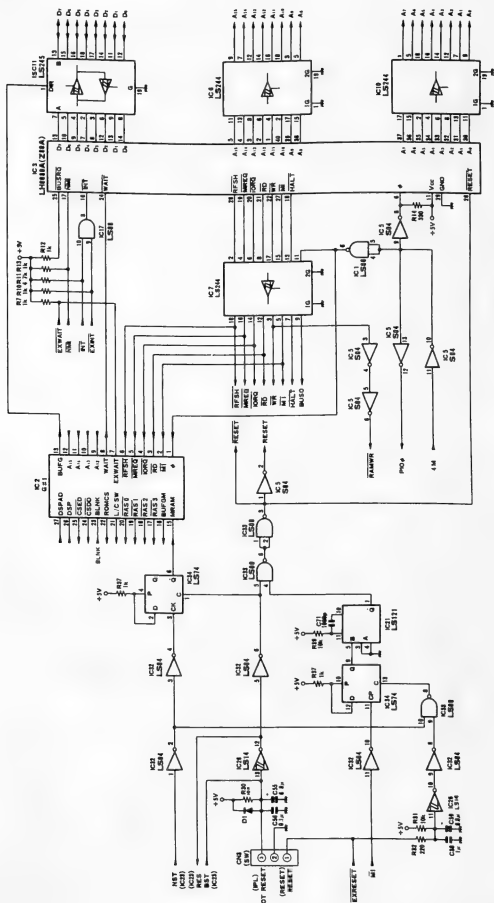
■拡張I/Oポートユニット, MZ-8BK(オプション)の回路図..... 28

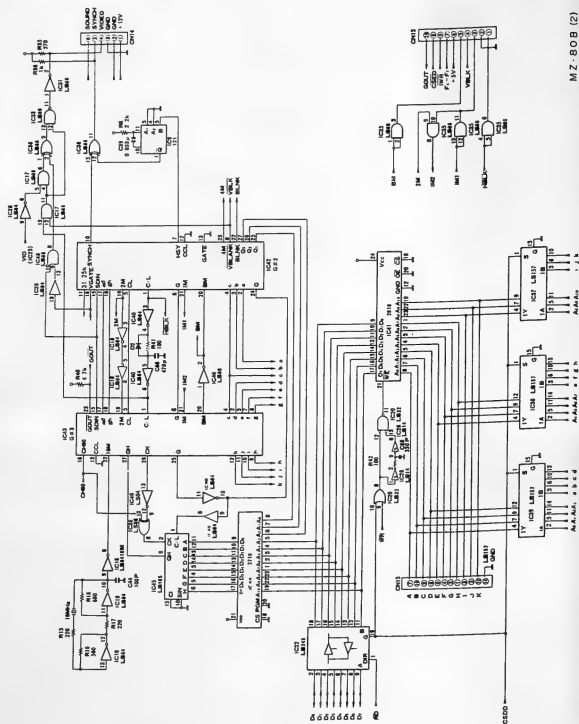
■グラフィックメモ리카ード2, MZ-8BGK(オプション)の回路図..... 29

MZ-80Bの全回路図やIPLプログラム、モニタは、既に本文中で紹介しているとおり、シャープの『MZ-80B OWNER'S MANUAL』や『OWNER'S MANUAL SUPPLEMENT』などで公開されています。

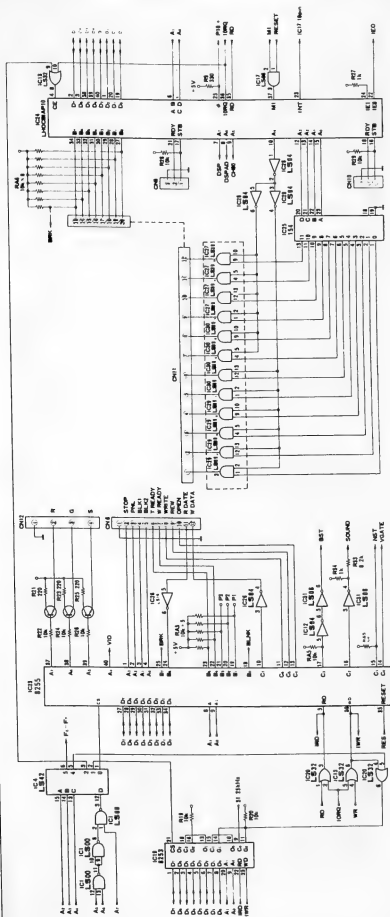
本回路図は『MZ-80B OWNER'S MANUAL』に掲載されている回路図を基にして編集部で作成したものです。したがって、この回路図に関して、メーカーに直接問い合わせることはご遠慮ください。また、正確な図面が必要な方は、メーカーの各マニュアルを参照してください。

■CPUボード〔1〕：CPU信号系





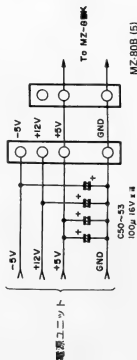
■CPUボード・ブロック(3): 8255, PIO信号系 ■CPUボード(5): CN4, CN, 電源の各コネクタ



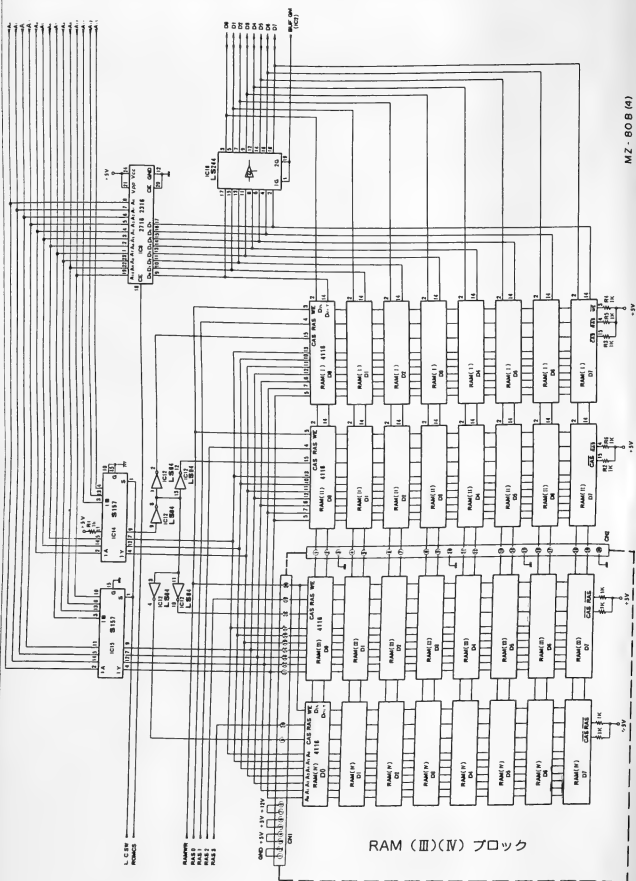
CN4 → MZ-8BK
CN5 → MZ-8B6

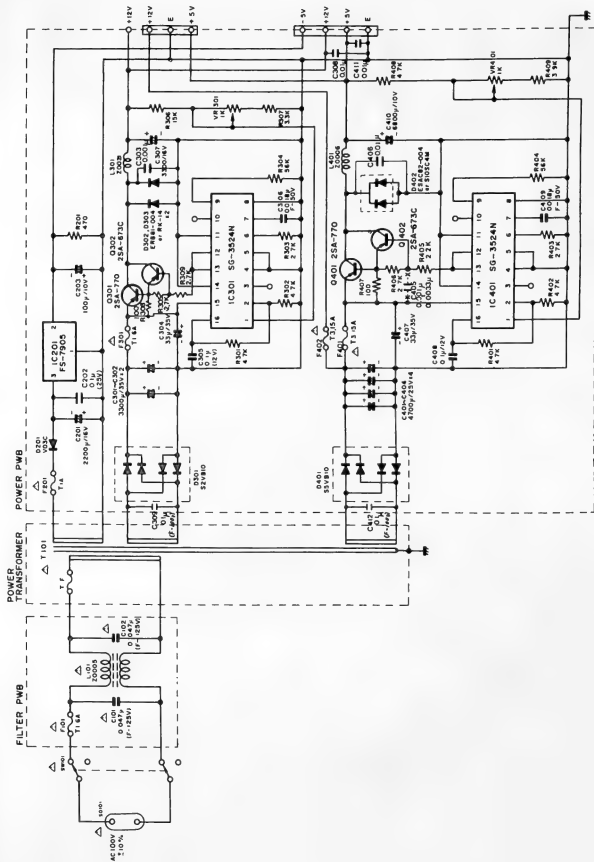
CN4, 5 40P			
1	A15	2	A14
3	A13	4	A12
5	A11	6	A10
7	A9	8	A8
9	GND	10	A7
11	A6	12	A5
13	A4	14	A3
15	A2	16	A1
17	A0	18	GND
19	D7	20	D6
21	D5	22	D4
23	D3	24	D2
25	D1	26	D0
27	GND	28	NMI
29	EX WAIT	30	EX INT
31	EX RESET	32	RESET
33	IO	34	HALT
35	MREQ	36	TREQ
37	RD	38	WR
39	MT	40	BUS

BUS CONNECTOR

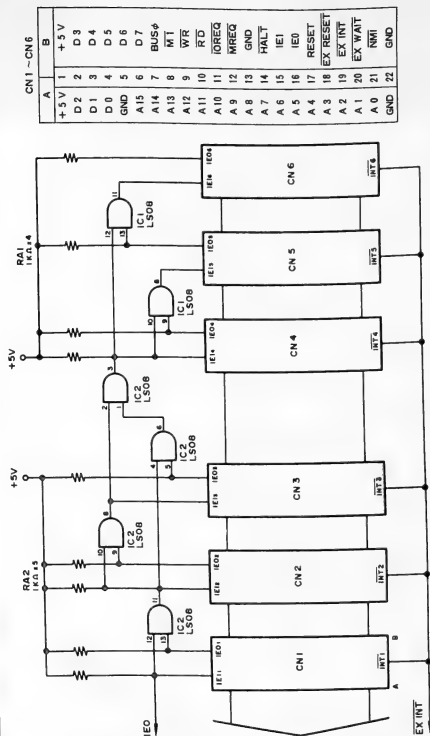


■CPUボード(4) : RAM回路





■拡張I/OポートユニットMZ-8BK(オプション)



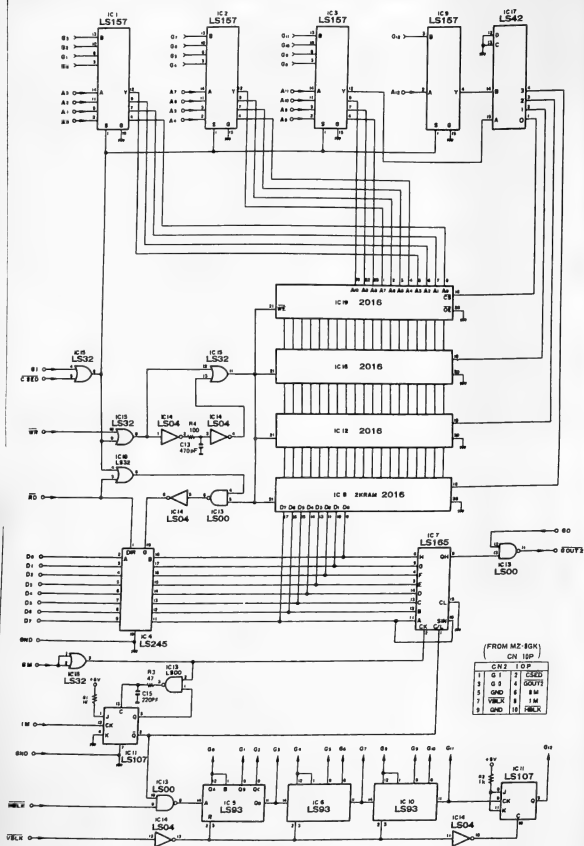
(To CPU BOARD)
CN4 40P

CN 7, 8 40P	
1	A15
2	A14
3	A13
4	A12
5	A11
6	A10
7	A9
8	A8
9	A7
10	A6
11	A5
12	A4
13	A3
14	A2
15	A1
16	GND
17	D7
18	D6
19	D5
20	D4
21	D3
22	D2
23	D1
24	D0
25	$\overline{\text{NMI}}$
26	GND
27	$\overline{\text{EX WAIT}}$
28	$\overline{\text{EX RESET}}$
29	$\overline{\text{EX INT}}$
30	RESET
31	HALT
32	$\overline{\text{IOREQ}}$
33	$\overline{\text{IEO}}$
34	$\overline{\text{MREQ}}$
35	$\overline{\text{RD}}$
36	$\overline{\text{WR}}$
37	$\overline{\text{MT}}$
38	BUS ϕ
39	
40	

CN1 ~ CN6

A		B	
1	+5V	1	D3
2	D2	2	D4
3	D1	3	D5
4	D0	4	D6
5	GND	5	D7
6	A15	6	BUS ϕ
7	A14	7	$\overline{\text{MT}}$
8	A13	8	$\overline{\text{WR}}$
9	A12	9	$\overline{\text{RD}}$
10	A11	10	$\overline{\text{IOREQ}}$
11	A10	11	$\overline{\text{MREQ}}$
12	A9	12	GND
13	A8	13	HALT
14	A7	14	$\overline{\text{IEI}}$
15	A6	15	$\overline{\text{IEO}}$
16	A5	16	$\overline{\text{RESET}}$
17	A4	17	$\overline{\text{EX RESET}}$
18	A3	18	$\overline{\text{EX INT}}$
19	A2	19	$\overline{\text{EX WAIT}}$
20	A1	20	NMI
21	A0	21	GND
22	GND	22	

■グラフィック・メモリ〔2〕：MZ-8BGK(オプション)



モニタの活用法	COMPAC	30
▶		
FORTAN-MZB	COMPAC	33
WICS 80B モニタ「TS-1000」	キャリアラボ 佐々木哲哉	47
WICS 80B インタープリタ	キャリアラボ 佐々木哲哉	53
WICS 80B コンパイラ	キャリアラボ 山本耕司	61
▶		
エディタ・アセンブラ「EDAS」	石橋義徳	73
ラベル付逆アセンブラ	COMPAC	85
SELF RELOCATABLE DEBUGGER	COMPAC	91
TRACER 80B	HUDSON SOFT 竹部隆司/中本伸一	95
▶		
SB-5520の拡張	山川孝文	101
日本語エラーメッセージ	佐々木雄一	112
COMMAND UP	ハゲミツ	118
SB-5520ユーティリティ・プログラム	亀谷英行	120
▶		
テープ・フォーマットの変換法	大垣泰二	127
PC→MZB CONVERTER	FALCON. S	128
MZB→PC CONVERTER	柴田文彦	130
MZ→MZBマシン語CONVERTER	野口岳郎, I/Oエンジンルーム	132
▶		
VOICE SYSTEM 80B	三屋一彦	134
BASIC VOICE	松井洋介	138
MUSICREATE	柴田文彦	145
▶		
グラフィック・キャラクタ・ジェネレータ	柴田文彦	148
PATTERN EDITOR	田中 聡	154
キャラクタ・グラフィック同時コピー	藤原秀樹	157
プラネタリウム	白木善喜	163
株価・出来高チャート	井上信夫	169
AUTO-2001「CMT-1, CMT-3」	宇部鉄工マイコンクラブ 縄田 悟	175
AUTO-2001「CMT-2」	宇部鉄工マイコンクラブ 縄田 悟	193
TV黒板	高嶋 晃	204
▶		
タテ・ヨコ チェック・サム		46
横8バイトのチェック・サム		52

モニ

●コムバック

MZ-80Bの場合、通常、モニタといえばSB-1520を指します。このモニタはマシン語の打ち込み、表示、テープ入出力をサポートしているわけですが、別の考え方をすると、それらを行なうマシン語サブルーチンを持っているということです。実際このモニタはサブルーチンの宝庫で、キーボード入力、CRT出力、テープ入出力、ASCIIコード→16進数、カセット・コントロール、サウンド・コントロール、画面コントロールなどがあります。

実際のマシン語プログラムを作る場合は、これらのサブルーチンを使えば基本的なこと（キー入力、画面表示）は簡単にできてしまいます。

サンプル・プログラムについて

リスト1を見てください。これは“Hello, Z”というメッセージを表示するものです。DEレジスタに文字列の先頭番地をセットしてモニタのMSG（0889H）を呼び出すことで処理しています。この場合、注意することは文字列の終わりが0D_Hになっていることで、このMSGは0D_Hが出てくるまで表示を続けるので、もし0D_Hがないと永久に表示し

続けます。また、レジスタの保存状況など、サブルーチン・コールをする場合、必要な情報はセットし、破壊するレジスタは保存するという手順を踏む必要があります。それさえ守ればこんな便利なものはありません。

リスト2、3、4はそれぞれアドレス入力、アドレス表示、データ表示となっていて、リスト2はキーボードからアドレス入力を行ない、HLレジスタに値をセットします。リスト3はHLレジスタにセットされた値を16進4桁で表示します。リスト4はHLレジスタで指定された番地のメモリの内容を表示します。リスト5はこの3つのプログラムを組み合わせたものです。

リスト6は1文字入力ルーチンです。モニタのSB-1520にはカーソルを点滅させながらキー入力待ちをするサブルーチンがありません。これはそれを行なうサブルーチンで、メモリ上のどこにあっても動作します。Aレジスタ以外は破壊しません。キーが押されるまで待つて、押されるとそのキーのASCIIコードをAレジスタにセットして戻ります。

みなさんもモニタ・サブルーチンを大いに活用してプログラムを作ってください。

1 文字出力（カーソル・コントロール可）

番地	08C5H
入力	Acc=ASCIIコード
出力	なし
レジスタ	AF以外は保存

1 文字入力（入力待ちでない）

番地	0832H
入力	なし
出力	Acc=ASCIIコード（キーを押さなければAcc=0）
レジスタ	AF以外は保存

CMTデータ書き込み

番地	0282H
入力	1140H～データ・バッファ（128バイト）
出力	CF=1：エラー発生
レジスタ	AF, BC, DE, HL以外は保存

1 文字出力（カーソル・コントロール不可）

番地	08B5H
入力	Acc=ASCIIコード
出力	なし
レジスタ	AF以外は保存

1 行入力

番地	06A4H
入力	DE=入力データの格納アドレス（06A2H）=入力可能な最大の文字数
出力	なし
レジスタ	全レジスタ保存

CMTファイル・インフォメーション・ブロック読み込み

番地	028EH
入力	なし
出力	CF=1：エラー発生 CF=0：エラーなし、エラーなし時、1140H以降にファイル・インフォメーション
レジスタ	AF, BC, DE, HL以外は保存

文字列出力（カーソル・コントロール可）

番地	0889H
入力	DE=文字列先頭（0Dで終わること）
出力	なし
レジスタ	全レジスタ保存

改行

番地	0A2EH
入力	なし
出力	なし
レジスタ	AF以外は保存

CMTデータ読み込み

番地	02B2H
入力	1140H～インフォメーション・バッファ
出力	CF=1：エラー発生
レジスタ	AF, BC, DE, HL以外は保存

文字列出力（カーソル・コントロール不可）

番地	087BH
入力	DE=文字列先頭（0Dで終わること）
出力	なし
レジスタ	全レジスタ保存

CMTファイル・インフォメーション・ブロック書き込み

番地	0251H
入力	1140～データ・バッファ（128バイト）
出力	CF=1：エラー発生
レジスタ	AF, BC, DE, HL以外は保存

CMTスペース・サーチ

番地	04B1H
入力	なし
出力	CF=1：エラー発生
レジスタ	CF=0：エラーなし（モータは止まらない） AF, BC, DE, HL以外は保存

CMTモータ停止

番地	04CEH
入力	ナシ
出力	ナシ
レジスタ	AF以外は保存

CMT早送り

番地	04E9H
入力	ナシ
出力	ナシ
レジスタ	AF以外は保存

CMT EJECT

番地	04E8H
入力	ナシ
出力	ナシ
レジスタ	AF以外は保存

ベル音(440Hz)

番地	0F14H
入力	ナシ
出力	ナシ
レジスタ	AF以外は保存

音楽テンポ設定

番地	0504H
入力	Acc = 1 ~ 7 (1が遅い)
出力	ナシ
レジスタ	全レジスタ保存

音楽演奏

番地	0F3FH
入力	DE = 音楽データの先頭 (0DHまたは2AHで終わること)
出力	CF = 0 : 演奏終了 CF = 1 : ブレーク・キーが押された。
レジスタ	AF以外は保存

音出し

番地	0F22H
入力	HL = 音程, BC = 音量
出力	ナシ
レジスタ	AF以外は保存

HEX→ASCII(CRT: 4桁出力)

番地	05D8H
入力	HL = 表示したい16進数
出力	CRTのカーソル位置より16進4桁で表示
レジスタ	AF以外は保存

HEX→ASCII(CRT: 2桁出力)

番地	05DDH
入力	Acc = 表示したい16進数
出力	CRTのカーソル位置より16進2桁で表示
レジスタ	AF以外は保存

HEX→ASCII(1桁)

番地	05F3H
入力	Acc = 下位4ビットに0~Fをセット
出力	Acc = 対応するASCIIコード
レジスタ	AF以外は保存

文字列比較

番地	063AH
入力	DE = "文字列1"の先頭アドレス HL = "文字列2"の先頭アドレス
出力	B = 比較する最大文字数 ZF = 1 : 一致した。 ZF = 0 : 一致しなかった。
レジスタ	AF以外は保存
備考	文字列中, 0DHで一致すればZF = 1とし比較を終わる。

ASCII→HEX(4桁)

番地	0614H
入力	DE = ASCII列先頭
出力	CF = 0 : HL = 16進数 CF = 1 : HLは不定
レジスタ	AF, HL以外は保存

ASCII→HEX(2桁)

番地	0623H
入力	DE = ASCII列先頭
出力	CF = 0 : Acc = 16進数 CF = 1 : Accは不定
レジスタ	AF, DE 以外は保存

ASCII→HEX(1桁)

番地	05FDH
入力	Acc = ASCIIコード("0"~"F")
出力	CF = 0 : Acc = 16進数 CF = 1 : Accは保存
レジスタ	AF以外は保存

画面→80字

番地	0C7CH
入力	ナシ
出力	ナシ
レジスタ	AF, BC, HL以外は保存

画面→40字

番地	0CEEH
入力	ナシ
出力	ナシ
レジスタ	AF, BC, HL以外は保存

リスト1

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
1	0000			ORG	9000H
2					
3	0089		MSG	EDU	08B9H
4	00B1		MONITOR	EDU	00B1H
5					
6	9000	110790		LD	DE,MSGG
7	9003	C0B908		CALL	MSG
8	9006	C3B100		JP	MONITOR
9					
10	9009	48556C4C	MSGG	DEFB	'Hello !!
	900D	AF72021			
	9011	21			
11	9012	0D		DEFB	0DH
12					
13	9013			END	
ERRORS 0					

リスト3

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
1					*****
2					18 PRINT OUT NOW ADDR *
3					*****
4					
5	08B9		MSG	EDU	08B9H
6	05DB		PRTHL	EDU	05DBH
7					
8	0000			ORG	8000H
9					
10	8000	110AB0		PRTADR:	LD DE,MSGADR2
11	8003	C0B908		CALL	MSG
12	8006	C0B905		CALL	PRTHL
13	8009	C9		RET	
14					
15	800A	4E4F5720		MSGADR2:DEFM	'NOW ADDRESS-*
16	800E	41444452			
17	8012	4553532D			
18	8016	24			
19	8017	0D		DEFB	0DH
20					
21	18	8018		END	
ERRORS 0					

リスト2

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS		
1					*****		
2					18 INPUT ADDRESS *		
3					18 ASCII→HEX TO HL *		
4					*****		
5							
6	06A4		GETL	EDU	06A4H		
7	08B9		MSG	EDU	08B9H		
8	0614		HLHEX	EDU	0614H		
9	0A2E		LETNL	EDU	0A2EH		
10	1093		BUFER	EDU	1093H		
11							
12	0000			ORG	8000H		
13							
14	8000	C02E0A		GETADDR:	CALL	LETNL	CR & LINE FEED
15	8003	111AB0		LD	DE, MSGADR	MSG	
16	8006	C0B908		CALL	MSG	LINE MESSAGE	
17	8009	119310		LD	DE, BUFER		
18	800C	C0A406		CALL	GETL	LINE INPUT	
19	800F	210F00		LD	HL, 15		
20	8012	19		ADD	HL, DE		
21	8013	EB		EX	DE, HL		
22	8014	CD1406		CALL	HLHEX	(DE) [ASCII]→HL[HEX]	
23	8017	5BEA		JP	CR	ERROR ! ONCE MORE AGAIN	
24	8019	C9		RET			
25							
26	801A	494E5055		MSGADR:	DEFB	'INPUT ADDRESS-*	
27	801E	5420A144					
28	8022	44524553					
29	8026	532024					
30	8029	0D		DEFB	0DH		
31							
32	802A			END			
ERRORS 0							

リスト5

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
1	06A4		GETL	EDU	06A4H
2	08B9		MSG	EDU	08B9H
3	0614		HLHEX	EDU	0614H
4	0A2E		LETNL	EDU	0A2EH
5	1093		BUFER	EDU	1093H
6	05DB		PRTHL	EDU	05DBH
7	05DD		PRTHX	EDU	05DDH
8	00B1		MONITOR	EDU	00B1H
9					

リスト 5

10	0000	ORG	8000H	
11				
12	8000	CD2E0A	GETADDR:CALL	LETNL ; CR & LINE FEED
13	8003	117580	LD	DE,MSGADR
14	8006	CD8908	CALL	MSG ; LINE MESSAGE
15	8009	19310	LD	DE,BUFER
16	800C	CD4A06	CALL	GETL ; LINE INPUT
17	800F	1A	LD	A,(DE)
18	8010	FE0B	CP	OBH ;
19	8012	CD8100	JP	Z,MONITOR ; CHECK BREAK KEY
20	8015	210F00	LD	HL,15
21	8018	19	ADD	HL,DE
22	8019	E8	EX	DE,HL
23	801A	CD1406	CALL	HLJEX ; (DE)[ASCII]+HL[HEX]
24	801D	38E4	JR	C,GETADDR+3 ; ERROR ! ONCE MORE AGAIN
25				
26	801F	114580	PRTADR: LD	DE,MSGADR2
27	8022	CD8908	CALL	MSG
28	8025	CD8805	CALL	PRTHL
29				
30	8028	115380	PRTCODE:LD	DE,MSGCD
31	8028	CD8908	CALL	MSG
32	802E	7E	LD	A,(HL)
33	803F	CD0005	CALL	PRTHX
34	8032	C3B100	JP	MONITOR
35				
36	8035	494E5055	MSGADR: DEFH	'INPUT ADDRESS-8'
	8039	54204144		
	803D	44524053		
	8041	532024		
37	8044	0D	DEFB	ODH
38				
39	8045	4E4F5720	MSGADR2:DEFH	'NDW ADDRESS-8'
	8049	41444452		
	804D	45535320		
	8051	24		
40	8052	0D	DEFB	ODH
41				
42	8053	20434F44	MSGCD: DEFH	'CODE-8'
	8057	452024		
43	805A	0D	DEFB	ODH
44				
45	805B		END	
ERRORS 0				

リスト 4

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
1					*****
2					18 PRINT (HL) *
3					*****
4					
5	0889		MSG	EDU	0889H
6	050D		PRTHX	EDU	050DH
7					
8	0000			ORG	8000H
9					
10	8000	110AB0	PRTCODE:LD	DE,MSGCD	
11	8003	CD8908	CALL	MSG	
12	8006	7E	LD	A,(HL)	
13	8007	CD8805	CALL	PRTHX	
14					
15	800A	20434F44	MSGCD: DEFH	'CODE-8'	
	800E	452024			
16	8011	0D	DEFB	ODH	
17					
18	8012		END		
ERRORS 0					

リスト 6

E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
1					*****
2					18 INPUT 1 KEY & CURSOR *
3					*****
4					
5	0832		GETKY	EDU	0832H
6	0C29		7PONT	EDU	0C29H
7	0C3E		DSPR	EDU	0C3EH
8	0C50		DSPW	EDU	0C50H
9	000D		FLASH	EDU	000DH
10	000E		FLSDAT	EDU	000EH
11					
12	0000	C5	??KEY:	PUSH	BC
13	0001	E5		PUSH	HL
14	0002	CD290C		CALL	7PONT ; GET VRAM ADDR TO HL
15	0005	CD3E0C		CALL	DSPR ; GET CHARACTER FROM VRAM (HL)+A
16	0008	320D00		LD	(FLASH),A
17	000B	320E00		LD	(FLSDAT),A
18	000E	3A0E00		LD	A,(FLSDAT)
19	0011	FE1F	KYJ1:	CP	1FH ; CHECK CURSOR CODE
20	0013	200B		JR	NZ,KYJ3
21	0015	3A0D00		LD	A,(FLASH)
22	0018	320E00	KYJ2:	LD	(FLSDAT),A ; SET NDW FLASHING CODE
23	001B	CD500C		CALL	DSPW ; SET CHARACTER TO VRAM A+(HL)
24	001E	1B04		JR	KYJ4
25	0020	3E1F	KYJ3:	LD	A,1FH
26	0022	1BF4		JR	KYJ2
27	0024	CD320B	KYJ4:	CALL	GETKY ; REAL TIME GET 1 KEY
28	0027	B7		DR	A
29	0028	20FA		JR	NZ,KYJ4
30	002A	010002		LD	BC,200H ; FLASH DELAY
31	002D	CD320B	KYJ5:	CALL	GETKY
32	0030	B7		DR	A
33	0031	2007		JR	NZ,KYJ6
34	0033	08		DEC	BC ; DELAY - 1
35	0034	7B		LD	A,B ; CHECK DELAY = 0
36	0035	B1		DR	C ;
37	0036	20F5		JR	NZ,KYJ5
38	0038	1B04		JR	KYJ1 ; JUMP LOOP
39	003A	F5	KYJ6:	PUSH	AF
40	003B	3A0D00		LD	A,(FLASH)
41	003E	CD500C		CALL	DSPW
42	0041	F1		POP	AF
43	0042	E1		POP	HL
44	0043	C1		PDP	BC
45	0044	C9		RET	
46					
47	0045			END	
ERRORS 0					



FORTAN-MZB

●COMPAC

I/O'81年1月号のFORTAN-MZをMZ-80Bに移植しました。グラフィック関係を少々強化しただけで、ほかはMZ-80K/C版と同じです。

コンパイラ

コンパイラ・モードで受け付けられるコマンドは11種類あります。

①EDIT

このコマンドは内蔵されているエディタにジャンプします。エディタの説明は後にします。

EDIT ☐ CR

②COMPILE

テキスト・ファイルがメモリ上にある場合に、そのテキストを読みながらコンパイルします。コンパイラによって展開されたオブジェクトはテキスト・エンドから作られています。

COMPILE ☐ CR

テキストがない場合、またはEND行がない場合には、

FILE ERROR ABORT!

がでます。

③EXEC

カセットにテキストが落ちている場合にこれを読みながらコンパイルすることができます。

EXEC (FILE NAME) ☐ CR

↓PLAY

このコマンドを実行すると、コンパイラはカセットから1ブロックごと読み込んで、そのつどコンパイルしていきます。

オンメモリで、コンパイルしたときにメモリエラーが出るような場合には、いったんカセットにテキストを落としてEXECで実行してください。

④BSAVE

COMPILEまたはEXECで作ったオブジェクトを外部へ出力するときにこれを使います。

BSAVE (FILENAME) ☐ CR



⑤LIST指定

COMPILE, EXECによりコンパイルすると、コンパイラはコンパイルした行を表示します。LIST指定によって全リストを表示したり、しなかったり、またエラーのあった行のみ表示することができます。

●LISTN

コンパイル・リストは表示せず、エラーの合計とメモリ・サイズを表示します。初期状態ではこのモードです。

●LISTE

エラーのなる行のみ出力して、エラー合計、メモリ・サイズを表示します。

●LIST

すべてのコンパイル・リストを出力します。

●LISTP

出力デバイスをプリンタに切り換えます。

LISTPはプリンタ・モードのLISTで、これの後にLISTN, LISTEを実行すると、プリンタ・モードのそれになります。電源が入っていない場合、UNDEFINEDを返してきます。

●LISTC

この指定によってプリンタ・モードを解除します。

⑥RUN

コンパイル・オブジェクトをFORTRANから直接実行する場合にRUNを使います。

RUNはオブジェクトを単に実行するだけで、その他の処理は行いません。

ただ、STOP文、BREAK文によって、オブジェクトが停止した場合のみコンパイラに戻るようスーパーバイズし

ています。

⑦BYE

モニタへジャンプします。

コールド・スタート	1300H
ホット・スタート	132AH

エディタ

エディタは、カーソル・エディットができる簡単なエディタです。画面上にリストが残っていれば、いつでも修正可能です。

メモリ内イメージは、シャープのTEXTエディタと同じく、CRで区切られた文字列です。カセットの記録フォーマットも同じなので、シャープのテキスト・エディタでソースを作って、コンパイルしてもかまいません。

コンパイラからエディタへジャンプします。

EDIT	CR
#	

これによりエディタ・モードを示すために#を出力して、入力待ちになります。

エディタ・コマンドは以下の8個です。

&...NEW D...DELETE
I...INSERT !...COMPILER
B...BREAK R...READ FILE
L...LIST W...WRITE FILE

①Insert

Iコマンドは、テキストをインサードするときに使います。このコマンドは常にテキスト・エンドヘインサードします。

[SHIFT] + [BREAK] を押すことによって、コマンド・レベルへ戻ります。

#I	CR
#	

②Break

Bコマンドは、行間に1行インサードしたいときに使います。

Bn	
----	--

nで指定した番号の行は1行下にさがり、その上にインサードされます。

#Bn	CR
Strings	CR
#	

nは、LISTしたときにエディタが付ける番号です。

③List

Lコマンドはインサードしたテキストをリストします。次の3つの型があります。

#L	CR
#L, n	CR
#L, P	CR

●L

テキストの先頭からCRTに表示します。[SHIFT]

[BREAK] によって途中で止めることが可能です。

●L, n

nはLISTしたときに、エディタが付ける番号で、n番以降のリストを行ないます。LISTを止めるのはLと同じです。

●L, P

プリンタにLISTを出力します。ただし、n番以降のLIST出力はできません。止める場合はLと同じです。

LISTの画面は次のようになっています。

#L	CR
0	: Strings
1	: Strings
2	: Strings
:	
#	

左側の番号がエディタの付けた番号で、Stringsはインサートしたテキストです。番号とテキストの間に『:』が入っています。

④Delete

Dコマンドは、1行デリートします。

#Dn	CR
-----	----

nで指定したテキストが抹消されます。

nはエディタの付けた番号です。

⑤Compile

!コマンドはエディタ・モードを抜け、コンパイラ・モードへ移る場合に使います。

#!	CR
OK.	
#	

⑥New

&コマンドはテキスト・エリアをクリアし、初期化します。

#&	CR
#	

⑦Read, Write

R, Wコマンドはカセットに対しての入出力用です。これによってテキスト・レベルでプログラムの保存ができます。

●R

テキストを読み込みます。

#R [FILE NAME]	CR
----------------	----

ファイル名は省略可能です。

●W

メモリにあるテキストをカセットに落します。

#W [FILE NAME]	CR
----------------	----

ファイル名は省略可能です。

⑧カーソル移動キー

画面にリストが残っていれば、BASICと同じようにカーソルを動かして修正が可能です。ただし、Iコマンド実行中ではありません。

カーソル・エディットするときは、エディタが付けた番号と“:”を消さないでください。これがなくなるとエディタはどのテキストを修正したかがわからなくなり、コマンド待ちの状態へ戻ります。

[CR] はどこで押してもかまいませんが、その行の下に

テキストがある場合には、";"の後にカーソルがなければ、#を出力してカーソルが点滅し、入力待ちになります。

このエディタは#がないとコマンドを受け付けられません。

FORTRAN-MZBの文法

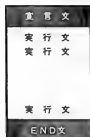
1) プログラムの構成

ソース・プログラムは図1のような構成になります。宣言文は省略可能です。END文はプログラムの最後に必要です。コンパイラはこれを見て、コンパイルを終了します。

各々の文は1行(1行は80字以内)に書かれ、継続はできません。文の1桁目に空白以外の文字があると、その文は注釈文になります。

文番号は必要な文にだけ、1~65,535の任意の数を割り当てただけで良く、BASICのようにすべての文に順に割り当てる必要はありません。

図1 ソース・プログラムの構成



2) 主プログラムと副プログラム

主プログラムと副プログラム(サブルーチン)はプログラム単位として区別されるのではなく、両者は1つのプログラムの中にあります。変数名は両者で共用されます。

3) 文の要素

①定数

a) 整数型定数

-32,767 ~ +32,767の10進数、あるいは\$0000 ~ \$FFFFの16進数

【例】 123, -65, \$0F, \$A12

b) 実数型定数

1.67×10^{-19} ~ 1.67×10^{18} の任意の数で有効桁数は7桁

【例】 1.0, -1.23, 1.5E+12, -0.2E-5

c) 文字型定数

その文字のASCIIコードが数値になる。

【例】 #A, #B

d) π π は3.14159と書くのと同じことを意味する。

②変数

変数名には1~4文字の一連の英数字を使います。最初の文字は必ず英字であることが必要です。変数名の最初の文字がI, J, K, L, M, Nのいずれかである場合は整数型の変数名、それ以外の英字である場合は実数型の変数名として区別されます。予約語は変数名として使われません。

③配列

配列はDIMENSION文で宣言します。配列名の付け方は変数の場合と同じで、やはり整数型と実数型に区別されます。

【例】 DIMENSION A(M) ただし、 $1 \leq M \leq 2,047$

DIMENSION K(M, N) ただし、 $1 \leq M \leq 255$

$1 \leq N \leq 255$

$1 \leq (M \times N) \leq 2,047$

④配列の添字

添字は、整数定数、整数変数、整数型算術式のいずれでもかまいません。

4) 算術式

算術演算子には、

- 1) \wedge (べき乗)
- 2) $*$, $/$ (乗算、除算)
- 3) $+$, $-$ (加算、減算)

の5種類があります。1), 2), 3)の順に優先度が高く、1つの算術式の中に多種類が使われた場合には、優先度の高いものから順に実行され、同一順位の演算子は左から右に実行されます。

べき乗にはJ \wedge I, A \wedge I, A \wedge Bの3種類の型があります。

J \wedge I型	J, Iともに整数型で、結果も整数型です。 $I \geq 0$ です。I = 0のとき、結果は1です。
A \wedge I型	Aは実数型、Iは整数型(Iは必ず整数型変数で、"2", "3"のような数を使うと次のA \uparrow B型になる)で結果は実数型です。 $-128 \leq I \leq 127$ に制限されます。
A \wedge B型	A, Bともに実数型で、結果も実数型です。ただし、 $A > 0$ 。

A \wedge I形はAをI回加算する方法、A \wedge B型は自然対数を使って計算する方法がとられるので、通常はA \wedge I型の方が格段に速く計算できます。上にも書きましたが、A \wedge Iで計算を行いたいときは、必ずIに整数変数を使わなければならない。

5) 代入文

一般形は、

$v = al$

で、右辺での演算結果を左辺の変数に代入します。左辺のvは変数、配列、メモリ関数、I/O関数のいずれかです。

変数名および配列名が整数型の場合には右辺の演算は整数型演算、実数型の場合には右辺は実数型演算として処理されます。

6) 制御文

①GOTO文

GOTO x

xは文番号。

②IF文

一般形は次のとおりです。

IF(al) x, y, z

()の中に書かれた算術式alの演算結果の負、零、正に基づいて、それぞれ、文番号x, y, zにジャンプします。算術式は"("の直後の文字が、A, B, ..., H, O, P, ..., Zの場合だけ実数型演算として処理され、それ以外の場合には整数型演算とみなされます。また、()の中に論理関数が使われた場合は整数型演算と同じです。

x, y, zのいずれも省略可能です。省略された場合に

は次の文にジャンプします。

③DO文

一般形は、

DO x v = n, m, l

xはループの端末文の文番号、vは整数です。nは開始値、mはループの終了値、lは増分のパラメータです。l、m、nは整数、整数、整数型算術式が可能です。lが1の場合は省略できます。

$l \geq 1, m \geq n \geq 1$

a) DO文は6重までネスティングが可能です。

b) DO文の範囲から外へ飛び出せるが、外からは飛び込めません。

④CONTINUE文

この文はプログラム中、どこにあって何も実行しませんが、DOの端末文、GOTO、IFなどの飛び先としても使えます。

⑤CALL文

この文は指定文番号を、副プログラムの先頭とみなしてコールします。変数は主プログラムと共用します。

⑥RETURN文

CALL文で呼ばれた副プログラムの最後に書き、この文が実行されるとCALL文の次の文へ戻ります。

⑦PAUSE文

この文を実行するとプログラムはそこで一時停止します。この状態でキーを押すと、次の文から実行を再開します。

例 PAUSE n

nはプログラマーが任意に付ける数字です。この文が実行されると「PAUSE n」が出力され、一時停止します。nはこのPAUSE文を知らせるための番号であるため、任意に付けてかまいません。必要なければnは省略可能です。

また、停止した状態では、**BREAK** キーが受け付けられます。

⑧STOP文

この文はプログラムの実行を停止させるときに使用します。STOP文はプログラム中どこに入れてもかまいません。

例 STOP n

nの関係はPAUSE文と同じです。

⑨BREAK文

BREAK キーのチェックのための文です。**BREAK** キーが押されなければ、何も実行しません。この文は端末文としても使用可能である。

⑩USR文

この文はマシン語とリンクするとき使用します。

例 USR (exp)

expで指定されるアドレスをサブルーチン・コールします。各レジスタ (AF, BC, DE, HL) はプログラム中で決めることができ、RETURNしたときのレジスタへの受け渡しはメモリを介して行なうので、MEM関数によってセットします。

各レジスタのアドレスは以下の通り。

AF	→12F0
BC	→12F2
DE	→12F4
HL	→12F6

ただし、AF, BC, DE, HL, IX, IY, SPを変えることはできません。

⑪END行

この行はソースの終わりを示すのでコンパイラはこの行を見つくとコンパイルを停止します。この文には文番号を付けることができません。

7) 入出力文

①SETG, RESG文

320×200の疑似グラフィックのセット、リセット文です。

例 SETG (exp1, exp2)
RESG (exp1, exp2)

exp1, exp2で指定される位置にオペレートします。exp1は横方向で0~319まで、exp2は0~199までの範囲です。exp1~2は定数、変数、式です。

②CHM

VRAM (キャラクターRAM) をクリアします。

③GRCLR

G-RAM 1 (グラフィックRAM 1) をクリアします。

④GRON

G-RAM 1をCRTに出力します。

⑤GROFF

G-RAM 1のCRT出力をやめます。

⑥READ

一般形は、

READ (v1 fmt, v2 fmt, ……)

vは変数名、配列要素で、fmtはその形式を表わします。fmtには次のようなものが入ります。

v.	10進整数入力
v. I	10進 "
v. B	16進 "
v. E	10進実数入力
v. A1	1文字入力
v. A2	2文字入力
"Strings"	* *の中を出力
	改行

10進整数入力の場合は「I」が省略できます。vは実数型変数名です。実数型入力の場合は、E型式、F型式のどちらでもかまいません。

ここで、E型式というのは、

1.0E+2, -2.30E-3

のように「E」を使って表わす数、F型式というのは、

1.0, -2.3, 0.123

のように「E」を使わないで表わす数の表現形式のことです。

⑦WRITE

一般型は、

WRITE (exp1, fmt, exp2, fmt, ……)

で、expは定数、変数名、式などです。

exp	10進左詰め整数表示
exp. In	n桁の10進右詰め整数表示
exp. E	E型式による実数表示
exp. B2	16進2桁表示
exp. B4	16進4桁表示

exp. X	expだけスペース表示
exp. A1	1 文字出力
exp. A2	2 文字出力
exp. V	カーソル・バーチカル・セット
exp. H	カーソル・ホリゾンタル・セット
"strings"	文字列出力
/	改行

expは定数、変数、式などです。

なお、0.A1でCRTをセット、1.A1でプリンタをセットすることができず。

各要素は「」で区切ることができるが、/ (改行) だけは区切る必要がありません。

8) 組み込み関数

組み込み関数は以下のとおりです。

- | | |
|-----------------------|-----------------------|
| 1) MEM (exp) | 12) COS (fxp) |
| 2) GET | 13) TAN (fxp) |
| 3) IOC (exp) | 14) EXP (fxp) |
| 4) LOW (exp) | 15) ALOG (fxp) |
| 5) MOD (exp1, exp2) | 16) ATAN (fxp) |
| 6) IRND (exp) | 17) FLOAT (fxp) |
| 7) IABS (exp) | 18) IFIX (fxp) |
| 8) ISIGN (exp1, exp2) | 19) IOR (exp1, exp2) |
| 9) ABS (fxp) | 20) IAND (exp1, exp2) |
| 10) SQRT (fxp) | 21) IXOR (exp1, exp2) |
| 11) SIN (fxp) | |

expは整数型の定数、変数、算術式を、fxpは実数型のそれらを表わしています。

MEM

MEM関数はexpで指定されたメモリをアクセスします。

v = MEM (exp)
MEM (exp) = exp1

MEMを右辺に置くと左辺に置くことによって機能が異なります。前者はPEEK的、後者はPOKEの機能を持ちます。

POKE機能の場合、exp1は0~255まで使用でき、それを越える場合には上位8ビットは無視されます。

GET

キーからリアルタイム入力を行ないます。GET実行中、キー入力があれば対応するASCIIコードの値を持ちます。入力がなければ0を返します。

例 v = GET
v = GET + exp

IOC

I/Oポートをアクセスします。

v = IOC (exp)
IOC (exp) = exp1

使用方法はMEMと同じです。

LOW

例 LOW (exp)

expの下位バイトを取った値を持ちます。expは-32,767~32,767までで、この関数が持つのは0~255までです。

MOD

MODは除算時の余りを与えます。

例 MOD (exp, exp1)

余りはexp/exp1のそれです。

IRND

この関数は乱数を与えます。

例 IRND (exp)

乱数は0~exp-1までの数です。expの上限は32,767までです。

IABS

例 IABS (exp)

この関数はexpの絶対値を与えます。

ISIGN

例 ISIGN (exp1, exp2)

exp1にexp2の持つ符号を与えます。この関数が持つ値はexp2の符号の付いたexp1である。

絶対値

ABS (fxp)

fxpは定数型の定数、変数、あるいは算術式 (以下、同じ) で、fxpの絶対値を与えます。

平方根

SQRT (fxp)

fxpの平方根を与えます。

三角関数

SIN (fxp)
COS (fxp)
TAN (fxp)

fxpの単位はラジアンで、それぞれ正弦、余弦、正接の値を与えます (180° = π ラジアン)。

指数関数

EXP (fxp)

指数関数値 e^{fxp} を与えます。

自然対数

ALOG (fxp)

自然対数 $\log_e (fxp)$ を与えます。

逆正接

ATAN (fxp)

逆正接 $\tan^{-1} (fxp)$ を与えます。

実数変換

FLOAT (exp)

整数型から実数型への変換用関数で、整数expの実数値を与えます。

【例】FLOAT(1) は 0.100000E+1
FLOAT(-2) は -0.200000E+1

整数変換

IFIX (fxp)

実数型から整数型への変換用関数で、実数fxpに対して、その整数値を与えます。

【例】IFIX(1.2345) は 1
IFIX(-1.23E+1) は -12

① IOR

IOR(*exp1*, *exp2*)

*exp1*と*exp2*との対応する各ビットごとにORを取った値を与えます。

【例】 IOR(\$0F, \$F0) は\$FF
IOR(\$1F, \$01) は\$1F

② IAND

IAND(*exp1*, *exp2*)

*exp1*と*exp2*との対応する各ビットごとにANDを取った値を与えます。

【例】 IAND(\$0F, \$F0) は\$00
IAND(\$1F, \$01) は\$01

③ IXOR

IXOR(*exp1*, *exp2*)

*exp1*と*exp2*との対応する各ビットごとにXORを取った値を与えます。なお、NOTは*exp1*か*exp2*のどちらかを\$F Fにすると得られます。

【例】 IXOR(\$0F, \$F0) は\$FF
IXOR(\$1F, \$01) は\$1E
IXOR(\$1F, \$FF) は\$E0

以上がFORTRAN-MZBの持つ関数であり算術式の中で自由に使えます。

式においては、*, /が+, -より先に演算され、()が付いているときのみそれが優先されます。()は6重まで許されます。

エラーメッセージ

コンパイラは次のようなエラーコードを表示します。

ERROR n

nはエラーコードで数字が出力されます。

①ERROR2

パラメータ誤り、または配列の添字の誤りがある場合、

②ERROR3

文番号の2重定義の場合。

③ERROR4

DIMENSION文の出でるところがおかしいとき。

④ERROR5

文法上のエラーの場合。

⑤ERROR6

DOの多重ループが6重以上となっている場合。

⑥ERROR7

変数名がおかしい。たとえば4文字以上の名前を付けたとき。

⑦ERROR8

DIMENSION内でのエラー。宣言時に2,047以上の添字を付けたとき。

⑧ERR DO LOOP

DOループがおかしいとき。

⑨MEMORY SINE OVER ABORT

メモリ・サイズをオーバーしたとき。

⑩ST NO, NOT FOUND

飛び先番号がない場合。

これらに加えて、実数形演算時でのエラーメッセージとして、下の4種類が表示されます。

①O ERROR ××××

計算途中でオーバーフローが起こったとき。××××はそれが発生した絶対番地。

②U ERROR ××××

計算途中でアンダーフローが起こったとき。

③N ERROR ××××

計算途中で、負の数が許されない場所に負の数が発生したとき。たとえば、SQRT(*fxp*)の*fxp*が負の値になったとき。

④D ERROR ××××

除算実行時の除数に異常が生じたとき。

プログラム例

1) 高速フーリエ変換

FORTRANの使える計算機では、高速フーリエ変換(FFT)はごく一般的なものになっており、音声認識、図形処理の強力な武器として広く活用されています。ここでは、FORTRAN-MZBのサンプル・プログラムとして、FFTプログラムを紹介します。

●プログラムの構成

A) 行番号0~28: 入力次元数*N*と入力の実数部、虚数部、XR(I), XI(I)を与えて、FFTサブルーチンをCALLする。メイン・ルーチンです。

B) 行番号29~38: FFTの結果を表示するサブルーチンです(CALL1000)。途中で表示を止めたいときは、任意のキーを押してください。表示終了後は、任意のキーが押されるまで止まっています。

行番号35~52: COSのテーブルを作成し、C(I)に代入するサブルーチンです(CALL5000)。

行番号53~134: FFTを実行するサブルーチンです(CALL5100)。

サブルーチン [5100] ではIPS=0のときFFT、IPS=1のとき逆FFTを計算します。整数と実数ではその演算時間に格段の差があるので、できるだけ実数演算を減らしました。この例(*N*=128)では、*FFT WORKING*が表示されてから、*FFT FINISHED*が表示されるまでに16秒を要します。

メイン・ルーチンPrg. 1, Prg. 2……を作成しておき「上述のB)の部分、Prg.1」あるいは、「上述のB)の部分、Prg.2」の順にカセットからロードすると、いろいろのメイン・ルーチンから、FFTサブルーチンをCALLすることができます。

FFTの例題プログラム

```
0:00000000000000000000000000000000
110
210      FFT-TRANSFORM      0
310
410      FORTRAN SAMPLE PROGRAM  0
510
6:00000000000000000000000000000000
71C
81      DIMENSION XR(512), XI(512), C(512)
91      N=128
101C   INPUT
111      WRITE(/" INPUT"/, " I", 7.X, " REAL")
121      WRITE(6.X, " IMAGINARY")
```

```

13: DO 10 I=1,N
14:   XR(I)=FLOAT(I-1)
15:   XI(I)=0
16:   WRITE(/,I,14,XR(I),E,XI(I),E)
17: 11 IY=GET
18:   BREAK
19:   IF(IY)11,,11
20: 10 CONTINUE
21: C
22:   CALL 5000
23:   IPS=0
24:   CALL 5100
25:   CALL 1000
26:   IPS=1
27:   CALL 5100
28:   CALL 1000
29:   STOP
30: C
31: 1000 WRITE(/," I",6,X,"REAL")
32:   WRITE(7,X,"IMAGINARY")
33:   DO 1020 I=1,N
34:     WRITE(/,I-1,15,XR(I),E,XI(I),E)
35: 1001 IY=GET
36:     BREAK
37:     IF(IY)1001,,1001
38: 1020 CONTINUE
39:   READ(IY,A1)
40:   RETURN
41: C
42: C COS-TABLE
43: 5000 N2=N/2
44:   P=2.0*/FLOAT(N)
45:   WRITE(/,"COS-TABLE MAKING")
46:   DO 5010 I=1,N/4+1
47:     C(I)=COS(P*FLOAT(I-1))
48:     IF(-I+1),5011,5011
49:     C(N-I+2)=C(I)
50: 5011 C(N2+1)=C(I)
51:     C(N2-I+2)=-C(I)
52: 5010 CONTINUE
53:   WRITE(/,"COS-TABLE COMPLETED")
54:   RETURN
55: C FFT-SUBROUTINE
56: C IF IPS=0, THEN FFT
57: C IF IPS=1, THEN INV-FFT
58: C N MUST BE GREATER EQUAL 4
59: 5100 WRITE(/,"FFT WORKING")
60:   N2=N/2
61:   N4=N/4
62:   ISW=1-2*IPS
63:   ID=1
64:   IE=N
65: 5180 CONTINUE
66: C WRITE(/,"IE=",IE,14)
67:   DO 5110 IA=1, ID
68:     IB=(IA-1)*IE
69:     DO 5110 J=1,IE/2
70:       M=(J-1)*ID*ISW
71:       K=IB+J
72:       L=K+IE/2
73:       BREAK
74: 5120 IF(M),5140,5130
75:       M=M+N
76:       GOTD 5120
77: 5130 IF(M-N)5140,,
78:       M=M-N
79:       GOTD 5130

```

```

80: 5140 V=C(M+1)
81:   NN=N4-M
82:   IF(NN),5150,5150
83:   W=C(NN+M+1)
84:   GOTD 5160
85: 5150 W=C(NN+1)
86: 5160 A=XR(K)-XR(L)
87:   B=XI(K)-XI(L)
88:   XR(K)=XR(K)+XR(L)
89:   XI(K)=XI(K)+XI(L)
90:   IF(M),5170,
91:   IF(M-N4),5171,
92:   XR(L)=A+V-B*W
93:   XI(L)=B+W+B*V
94:   GOTD 5110
95: 5170 XR(L)=A
96:   XI(L)=B
97:   GOTD 5110
98: 5171 XR(L)=-B
99:   XI(L)=A
100: 5110 CONTINUE
101:   ID=ID*2
102:   IE=IE/2
103:   IF(IE-1),,5180
104: C
105:   WRITE(/,"BIT REVERSAL")
106:   N1=N/256
107:   N2=N-N1*256
108:   MEM(*4CF0)=N2
109:   MEM(*4CF1)=N1
110:   DO 5190 I=1,N
111:     IO=I-1
112:     I1=IO/256
113:     I2=IO-I1*256
114:     MEM(*4CF2)=I2
115:     MEM(*4CF3)=I1
116: 99 *ML 2A,F0,4C,ED,5B,F2,4C,AF,47,CB,3A
117:     *ML CB,1B,CB,10,17,CB,3C,CB,1D,CB,45
118:     *ML 2B,F1,21,F4,4C,70,23,77
119:     J=MEM(*4CF5)*256+MEM(*4CF4)
120:     IF(IO-J),5190,5190
121:     AR=XR(I)
122:     AI=XI(I)
123:     XR(I)=XR(J+1)
124:     XI(I)=XI(J+1)
125:     XR(J+1)=AR
126:     XI(J+1)=AI
127: 5190 CONTINUE
128: 5191 IF(IPS-1),5200,
129:   WRITE(/,"FFT FINISHED")
130:   RETURN
131: 5200 DO 5210 I=1,N
132:     XR(I)=XR(I)/FLOAT(N)
133:     XI(I)=XI(I)/FLOAT(N)
134: 5210 CONTINUE
135:   WRITE(/,"INV-FFT FINISHED")
136:   RETURN
137:   END

```

2) リサージュ図形

グラフィック・コマンドを使ってリサージュ図形を描かせるプログラムです。

```

01: C
11: C GRAPHIC TEST
21: C -- リサージュ --
31: C
41: GRCLR
51: L=160
61: M=100
71: CHM
81: GRON
91: DO 10 I=1,1200,2
101: A=FLOAT(I)
111: X=SIN(A/180.0)
121: Y=SIN(A/45.0)
131: X=X*100.0
141: Y=Y*100.0
151: L1=IFIX(X)
161: M1=IFIX(Y)
171: L2=L+L1
181: M2=M+M1
191: SETG(L2,M2)
201: 10 CONTINUE
211: END

```

参考文献

- 1) 野沢, 竹部, 中本: "FORM", I/O, '80年5月号～9月号
- 2) 秋野 実: "FORTRAN-MZ", I/O, '81年7月号

★使用方法

プログラムをスタートすると、

"SATRT, END ADD?"と聞いてくるので、スタートとエンド・アドレスを入力してください。このときコマを忘れると受け付けません。アドレスを入力すると、次は、

"PRINTER ON ?(Y/N)"と聞いてきます。プリンタに出力する場合は **[Y]** **[CR]**、他のキーはすべて **[N]** と判断します。

チェック・サム プログラム 1

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
1300 CD 2E 0A 11 2C 15 CD 2E 0A CD 89 08 CD 2E 0A CD 18C
1310 2E 0A 11 3E 15 CD 89 08 CD 2E 0A 11 93 10 CD AA 124
1320 06 1A FE 0B CA 00 00 CD 14 06 38 E3 22 8E 15 7C 86
1330 32 C3 15 CD 8C 14 1A 13 FE 2C 20 D3 CD 14 06 38 1E0
1340 CE 22 CD 15 CD 8C 14 1A FE 00 20 D3 CD AE 14 30 1F9
1350 BE 2A BE 15 7D E6 F0 6F 22 8E 15 11 9B 15 08 AE 10F
1360 08 CD 2E 0A AF 32 C2 15 11 93 10 CD AA 06 1A FE 108
1370 59 20 08 3E 01 32 C2 15 CD 1B 15 CD 7C 0C 21 AD 189
1380 15 AF 0A 11 77 23 10 7C 5D 21 80 15 CD 92 14 11 1E5
1390 AF 15 CD 5A 14 CD 92 14 0A 10 2A BE 15 CD AS 14 1A7
13A0 11 AD 15 CD 31 1A C5 06 10 CD 32 08 FE 0B CA 0F 187
13B0 13 7E F5 CD 76 14 CD 31 14 F1 F5 D0 86 00 D0 77 10C
13C0 00 F1 EB 8A 77 EB 23 13 10 DF 22 8E 15 3E 3A CD 123
13D0 8A 14 D0 7E 00 CD 76 14 CD 45 14 D0 3A 00 00 C1 146
13E0 CD AE 14 30 02 10 B3 3E 20 06 38 CD 86 14 10 FB 19F
13F0 CD 45 14 11 89 15 CD 6A 14 11 AD 15 06 10 0E 00 117

```

Sum CB 35 AF DF C5 C1 45 DF 0C D2 6E 72 14 B1 CA B0 13B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
1400 1A F5 CD 76 14 CD 31 14 F1 B1 AF 13 10 F2 3E 3A 160
1410 CD 8A 14 79 CD 76 14 CD 45 14 CD 92 14 CD AE 14 15F
1420 F5 CD 32 08 57 28 FA F1 DA 7E 13 CD 14 0F C3 00 1E4
1430 13 3E 20 CD 86 14 C9 C5 3E 20 11 93 10 06 50 12 1E0
1440 13 10 FC C1 C9 FD 3A 00 00 11 93 10 CD 5A 14 CD 1A1
1450 37 14 CD 92 14 C9 C9 08 3A C2 15 87 28 0A 1A 1F9
1460 CD C9 14 1A 13 FE 00 20 F6 C9 1A 13 FE 08 28 05 126
1470 CD 8A 14 1B F5 C9 F5 E6 F0 0F 0F 0F 0F CD B1 E1 146
1480 F1 E6 0F CD F3 05 FD 77 00 FD 23 C9 21 04 00 19 144
1490 EB C9 CD 2E 0A 3A C2 15 87 28 05 3E 0A CD C9 1A 1F9
14A0 FD 21 93 10 C9 7C CD 76 14 7D CD 76 14 C9 3A 13 1F7
14B0 15 2A BE 15 87 28 07 7C 87 20 03 87 1B 0A ED 58 16F
14C0 00 15 A7 ED 52 32 C3 15 C9 F5 AF CD DE 14 F1 D3 185
14D0 FF 3E 8D 03 FE 3E 01 CD DE 14 AF D3 FE C9 C5 D5 16F
14E0 57 1E 0C 01 00 00 DB FE E6 00 BA 20 03 D1 C1 C9 186
14F0 08 7B B1 20 F1 10 2E EE D1 C1 CD 2E 0A AF 32 C2 18A

```

Sum E2 CD 35 4A C1 7C 5F 72 29 EF 9B 6E 19 2D 5F DE 1EF

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
1500 15 11 8F 15 CD 89 08 C3 0F 13 47 3E 18 CD C9 14 157
1510 78 CD C9 14 6F CD DE 14 0B FE 0F 0F C9 3E 03 CD 15E
1520 0A 15 30 B6 3E 04 CD 0A 15 30 CF C9 43 48 45 43 12E
1530 48 20 53 55 40 20 20 4D 5A 20 38 30 42 08 53 54 1D2
1540 41 52 54 2C 45 4E 44 20 41 44 44 20 3F 20 00 41 180
1550 44 44 20 28 30 20 28 31 20 28 32 20 28 33 20 1FA
1560 2B 34 20 28 35 20 28 36 20 28 37 20 28 38 20 28 180
1570 39 20 28 41 20 28 42 20 28 43 20 28 44 20 28 45 1FF
1580 20 28 46 20 3A 33 75 40 00 53 75 60 20 20 50 1FF
1590 52 49 4E 54 45 52 20 45 52 52 00 52 49 4E 54 177
15A0 45 52 20 4F 4E 20 3F 28 59 2F 4E 29 00 AD E3 4E 18A

```

Sum A2 E3 4E CF 99 08 7B A9 CE 14 F3 C9 B6 0E 2D 3B 12E

256バイト出力し終ると、キー入力待ちになります。何かキーを押す
再び続行します。メンドクサイ人は\$1420から\$1427(E120から\$E
までを00にしてください。すべて終わると、また"START, END AD
と聞いてきます。 **[BREAK]** キーを押すとモニタにジャンプします

★プログラム・サイズ

\$1300 - \$15AD

(\$E000 - \$E2AD)

スタートは\$1300 (\$E000) です。

チェック・サム プログラム 2

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
E000 CD 2E 0A 11 2C E2 CD 2E 0A CD 89 08 CD 2E 0A CD 159
E010 2E 0A 11 3E E2 CD 89 08 CD 2E 0A 11 93 10 CD AA 1F1
E020 06 1A FE 0B CA 00 00 CD 14 06 38 E3 22 8E 15 7C 133
E030 32 C3 E2 CD 8C E1 1A 13 FE 2C 20 D3 CD 14 06 38 17A
E040 CE 22 CD E2 CD 8C E1 1A FE 00 20 C3 CD AE E1 30 160
E050 BE 2A BE E2 7D E6 F0 6F 22 8E E2 11 9B E2 CD 89 1F0
E060 08 CD 2E 0A AF 32 E2 11 93 10 CD AA 06 1A FE 1D5
E070 59 20 08 3E 01 32 C2 E2 11 93 10 CD 10 E2 CD 7C 0C 21 AD 185
E080 E2 AF 0A 11 77 23 10 FC D0 21 80 E2 CD 92 E1 11 17C
E090 AF E2 CD 56 E1 CD 92 E1 0A 10 2A BE E2 CD AS E1 1A8
E0A0 11 AD E2 CD 31 E1 C5 06 10 CD 32 08 FE 0B CA 0F 143
E0B0 E0 7E F5 CD 7A E1 CD 31 E1 F1 F5 D0 86 00 DD 77 1F3
E0C0 00 F1 EB 8A 77 EB 23 13 10 DF 22 8E E2 3E 3A CD 1F0
E0D0 B6 E1 DD 7E 00 CD 76 E1 CD 45 E1 DD 3A 00 00 C1 1AD
E0E0 CD AE E1 30 02 10 B3 3E 20 06 38 CD 86 E1 10 FB 139
E0F0 CD 45 E1 11 89 E2 CD 6A E1 11 AD E2 06 10 0E 00 148

```

Sum 62 CF E3 79 5F C2 12 13 A6 D2 D5 0C AE 4B 2D BA 1DC

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
E100 1A F5 CD 76 E1 C5 06 10 E1 F1 B1 AF 13 10 F2 3E 3A 160
E110 CD 8A E1 79 CD 76 E1 C5 06 10 E2 92 E1 CD AE E1 160
E120 F5 CD 32 08 57 28 FA F1 DA 7E 00 CD 14 0F C3 00 1B1
E130 CE 20 CD 86 E1 C9 C5 3E 20 11 93 10 06 50 12 17A
E140 13 10 FC C1 C9 FD 3A 00 00 11 93 10 CD 5A E1 CD 14E
E150 37 E1 CD 92 E1 C9 CD 89 08 3A C2 E2 87 28 0A 1A 160
E160 CD C9 E1 1A 13 FE 00 20 F6 C9 1A 13 FE 08 28 05 1F3
E170 CD 8A E1 1B F5 C9 F5 E6 F0 0F 0F 0F 0F CD B1 E1 140
E180 F1 E6 0F CD F3 05 FD 77 00 FD 23 C9 21 04 00 19 144
E190 EB C9 CD 2E 0A 3A C2 E2 87 28 05 3E 0A CD C9 E1 13A
E1A0 FD 21 93 10 C9 7C CD 76 E1 7D CD 76 E1 C9 3A C3 191
E1B0 E2 2A BE E2 7D 7C 87 20 03 87 1B 0A ED 58 16F
E1C0 CD E2 AF ED 52 32 C3 E2 C9 F5 AF CD DE E1 F1 D3 11C
E1D0 FF 3E 8D 03 FE 3E 01 CD DE 14 AF D3 FE C9 C5 D5 13C
E1E0 57 1E 0C 01 00 00 DB FE E6 00 BA 20 03 D1 C1 C9 186
E1F0 08 7B B1 20 F1 10 2E EE D1 C1 CD 2E 0A AF 32 C2 18A

```

Sum 7C 76 9C 17 5B 49 2C D9 F6 89 6B 3B B3 FA 2C 45 1BE

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
E200 E2 11 8F E2 CD 89 08 C3 0F 0F 0F 0F 0F C9 3E 03 CD 18B
E210 78 CD C9 E1 AF CD DE 14 0B FE 0F 0F C9 3E 03 CD 1F8
E220 0A E2 30 D6 3E 04 CD 0A E2 30 CF C9 43 48 45 43 12E
E230 48 20 53 55 40 20 20 4D 5A 20 38 30 42 08 53 54 1D2
E240 41 52 54 2C 45 4E 44 20 41 44 44 20 3F 20 00 41 180
E250 A4 44 20 28 30 20 28 31 20 28 32 20 28 33 20 1FA
E260 2B 34 20 28 35 20 28 36 20 28 37 20 28 38 20 28 180
E270 39 20 28 41 20 28 42 20 28 43 20 28 44 20 28 45 1FF
E280 20 28 46 20 3A 33 75 40 00 53 75 60 20 20 50 1FF
E290 52 49 4E 54 45 52 20 45 52 52 00 52 49 4E 54 177
E2A0 45 52 20 4F 4E 20 3F 28 59 2F 4E 29 00 00 00 00 1E7

```

Sum 6F B0 4E 69 99 08 7B 7A 9B E1 F3 C9 B6 4C 4A BA 1C3

MZ-80B

WICS 80B

1 MONITOR TS-1000

キャリア・ラボ 佐々木哲哉

今回から、MZ-80B用のWICSを発表していきます。MZ-80B用のWICSにはグラフィック用のコマンド（CIRCLE、PAINTなど）が附加され、さらに使いやすくなっています。また、C-DOSを使えば、ディスクに対しても、ロード、セーブが行なえます。

今回はWICS 80Bを動かすために必要なモニタを発表します。このモニタはキー入力や画面出力、プリント出力などの各種サブルーチン群とメモリ・ダンプなどのモニタ・コマンドからなっています。

MONITOR TS-1000の特徴

- MONITOR TS-1000では、すべてのキャラクタをキーボードから入力できるようにしています。
- キー入力はすべてのキーにオート・リピートがきくようになっています。
- SHIFT LOCK** キーにコントロール・キーの機能をアップしてあります。
- キー入力により、プリンタの出力を切り換えられます。
- キー入力により、リストの繰り替えも可能です。
- メモリ・ダンプ後は、スクリーン・エディットにて、メモリ変更ができます。
- MONITOR コールは\$0028だけですみます。

★キー入力時の注意

- GRPH** キーを入力した後（グラフィック・モード）



- グラフィック・モードで、**SHIFT** を押したままで、カーソル・キーを入力。



以上のキャラクタがキー入力でも画面に出力できるようになっています。

MONITOR TS-1000の機能

MONITORプログラムは、MZ-80Bのハードウェア装置であるキーボード、CRTディスプレイ、カセットテープ・デッキ、タイマ、サウンド発生回路、プリンタなどのI/Oをコントロールするサブルーチンが含まれています。WICSはこのモニタ・サブルーチンを使うことによってコンパクトになっています。

このモニタは\$0000～\$1500の領域があります。

コマンドの説明

MONITOR TS-1000には次のようなモニタ・コマンドがあります。

- D** コマンド：メモリの内容を表示（プリント出力も可）
- J** コマンド：任意のアドレスにコントロールを移す、プログラムの実行。
- S** コマンド：メモリ内容をカセットにセーブする。
- L** コマンド：メモリにカセットからロードする。
- V** コマンド：カセットテープとメモリ内容を比較する。

D コマンド実行後のダンプはそのままスクリーン・エディットで変更できます。

★モニタ・コマンドの機能と使い方

- ① **D** コマンド（D：memory Dump）

機能：指定するメモリ・アドレスから16進出力とASCII出力を行います。

例 *D1200 **CR**

1200番地より、メモリ・ダンプします。**BREAK** でストップ。

- ② **S** コマンド（S：Save）

機能：指定するメモリ・ブロックの内容をファイル・ネームを付けてカセットにセーブします。

例 *S1200, 1300, 12A0, MON **CR**

この例は、\$1200～\$1300のプログラムに“MON”というファイル・ネームで\$12A0にオート・スタートするように、カセットにセーブします。

- ③ **L** コマンド（L：Load）

機能：指定したファイル名のファイルをメモリにロード

します。

例 *LMON CR

この例は、“MON”というファイルをサーチしてロードします。ファイル名を省略すると、最初に見つかったファイルをロードします。

OV コマンド (V:Verify)

機能：指定したファイル名のベリファイします。

例 *V MON CR

この例は“MON”というファイルをサーチしてベリファイします。ファイル名を省略すると最初に見つかったファイルをベリファイします。

●J コマンド (J:Jump)

機能：指定したアドレスにコントロールを移します。

例 *J1500 CR

この例は\$1500をCPUのプログラム・カウンタに与えます。\$1500～に“WICS”があれば、“WICS”が起動します。ユーザーのプログラムがあれば、それを実行することになります。

※Dコマンドを実行させ、**BREAK** キーでコマンド待ちにした状態で、スクリーン・エディットを使い、メモリ内容が変更できます。このスクリーン・エディタは、ASCIIコードを入力することもできます。

スクリーン・エディタから抜けるとき、**BREAK** キーを押します。

例 \$8000 00 "ABCD" 0D CR \$8006 C3 0000 CR

この例は\$8000から、\$00, “A”, “B”, “C”, “D”, \$0Dを書き、\$8006より\$C3, \$00, \$00を書きます。このように、このスクリーン・エディタは、メモリ書き込みコマンドの役も果たします。

コントロール・キー表

MONITOR TS-1000では、**SFT LOCK** キーをコントロール・キーとして使っています。コントロール・キーの入力は**SFT LOCK** を押したままで、A-Zのキーを入力します。次に、コントロール・キーの内容を表にします。

コントロール・キー表

コード	キー	内容	コード	キー	内容
00		NUL	10	P	PRINTER ON
01	A	↓ (DOWN)	11	Q	TAB 10
02	B	↑ (UP)	12	R	BELL
03	C	→ (RIGHT)	13	S	SCROLL UP
04	D	← (LEFT)	14	T	SCROLL DOWN
05	E	H (HOME)	15	U	TAKI ON
06	F	C (CLR)	16	V	TAKI OFF
07	G	DEL	17	W	CAP
08	H	INST	18	X	
09	I	LINE CLR	19	Y	
0A	J	ALL CLR	1A	Z	
0B	K	WIDTH40			BREAK
0C	L	WIDTH80			
0D	M	CRLF			
0E	N	CRLF2			
0F	O	PRINTER OFF			

コントロール・キー

コード	キー	内容	説明
00		NUL	何も行わない。
01	A	↓ (DOWN)	カーソルを1行下へ移動させます。カーソルが最下行にあればスクロール・アップします。
02	B	↑ (UP)	カーソルを1行上へ移動させます。カーソルが最上行にあれば、何もしません。
03	C	→ (RIGHT)	カーソルを1桁右へ移動させます。カーソルが最右桁にあれば、1行下の桁の左にカーソルをさせます。
04	D	← (LEFT)	カーソルを1桁左へ移動させます。カーソルが最左桁にあれば、1行上の桁の最右桁にカーソルを移動させます。
05	E	H (HOME)	カーソルをHOME位置に移動させます。
06	F	C (CLR)	スクロール・ウィンドウ内部をクリアし、HOMEを実行します。
07	G	DEL	カーソル位置以後の文字をすべて1桁左に移動させ、カーソルも左へ移動させます。もし、カーソルが最左桁にあれば、カーソルは移動しません。
08	H	INST	カーソル以後の文字を1桁右に移動させ、カーソル位置にスペースを置きます。
09	I	LINE CLR	カーソル以後の文字1行分クリアします。
0A	J	ALL CLR	スクロール・ウィンドウの画面に関係なく画面全体を、クリアします。
0B	K	WIDTH40	40桁モードにします。クリアは行いません。
0C	L	WIDTH80	80桁モードにします。クリアは行いません。
0D	M	CRLF	改行を行ないます。
0E	N	CRLF2	カーソルが最左桁以外にあれば、改行を行ないます。
0F	O	PRINTER OFF	プリンタの出力を行わないモードにします。
10	P	PRINTER ON	プリンタの出力を行なうモードにします。
11	Q	TAB10	10桁のタブレーションを行ないます。
12	R	BELL	ベルを鳴らします。
13	S	SCROLL UP	スクロール・アップを行ないます。
14	T	SCROLL DOWN	スクロール・ダウンします。このとき、消しモードであれば消しを行ないます。
15	U	TAKI ON	消しモードにします。
16	V	TAKI OFF	消しモードを解除。
17	W	CAP	英大文字、英小文字の切り換えを行ないます。

モニタ・サブルーチン

MONITOR TS-1000では入出力などの最少限必要な機能は、\$0028番地をコールすることですべて行なえるように設計されています。

\$0028番地をコールするにはRESTART(5)を行なうとよいわけです。RESTART(5)はマシン語コードでは、\$EFの1バイトです。この1バイト\$EFだけでは\$0028～のサブルーチンは何を行えばよいのか分かりません。

そこで\$EFの後に1バイトの入出力の機器番号を書きます。たとえば、\$EF\$00であれば、1行入力行なない、\$EF\$06は1行出力を行なうと決めておけば、ユーザーはモニタのアドレスを覚える必要がありません。\$EFの後の1バイトの機器番号さえ覚えておけば、マシン語でもBASEでも容易にプログラムができるわけです。

次に、機器番号と名称表を示します。

機器番号と名称表

00	GETLN	14	MUSIC	28	CURHL
01	CRTOU	15	TIMESET	29	A = (HL)
02	PUTCHR2	16	TIMERD	2A	(HL) = A
03	PUTCHR	17	BELL	2B	FILELOAD
04	PUTLN	18	TEMPO	2C	FILEPR
05	PUTLN2	19	PRHL	2D	WINDOWSET
06	PUTLINE	1A	PRA	2E	BACKSEARCH
07	CRLF	1B	RDHEX4		
08	CRLF2	1C	RDHEX2		
09	SPACE	1D	MOTOR		
0A	GETCH	1E	MCSTOP		
0B	getch	1F	COPEN		
0C	INKEY	20	SEARCH SPACE		
0D	BREAK	21	FF		
0E	FSAVE	22	REW		
0F	PSAVE	23	PRSP		
10	FLOAD	24	PRH2		
11	FLOAD	25	SAME		
12	VERIFY	26	3PRC		
13	SOUND	27	2PR		
					ユーザー用

WICS モニタ ダンプ・リスト

9060	AE	4E	A2	50	A3	40	A6	00	F5	C5	E6	0F	47	7E	08	90	143
9070	72	1D	00	C1	F1	C9	C5	32	0F	00	ED	53	16	00	7E	C1	125
9080	03	E5	7E	AB	D3	E5	7E	02	D3	E4	AF	D3	E4	D3	F0	7E	184
9090	44	D7	E7	DB	E5	AF	DB	E5	FE	AB	20	F3	3E	C0	B9	20	15D
90A0	EE	D3	E5	7E	AB	D3	E5	7E	12	D3	E4	7E	7A	D3	E4	7E	1FB
90B0	84	D7	E7	DB	E6	AF	DB	E6	47	ED	47	1E	00	C1	C9	C5	1F3
90C0	E5	7E	BA	D3	E7	7E	44	D3	E7	DB	E6	4F	DB	E6	47	DB	190
90D0	E5	7E	DB	E5	7E	2A	1E	00	B7	ED	42	70	0F	CD	1D	11	11F
90E0	05	7A	B3	20	03	11	C0	AB	21	C0	AB	ED	52	ED	5B	16	1C4
90F0	00	19	7B	27	E5	11	C0	AB	ED	52	7B	14	F1	EB	7A	0F	1B2
Sum	27	B3	0A	73	46	5A	0B	DF	B2	4A	9C	DB	17	AC	9C	90	1CD

add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
9100	00	EE	01	E1	01	01	00	ED	42	20	02	EE	01	E1	C1	C9	17D
9110	D1	E1	3A	0F	00	1B	F6	11	40	57	19	1B	E0	ED	47	1E	110
9120	00	7A	0F	00	EE	01	32	0F	00	C9	C5	E5	F5	01	30	00	112
9130	21	60	00	EF	17	F1	E1	C1	C9	C5	05	7E	05	CD	40	11	1E7
9140	7E	04	CD	4D	11	0B	7B	B1	20	F1	D1	C1	C9	D3	E3	5A	117
9150	50	1B	7A	B3	20	F8	C9	C5	05	7E	05	7E	02	72	20	00	1A1
9160	FE	0F	DB	2C	FE	57	2A	1E	00	B7	ED	42	70	0F	CD	1D	11F
9170	21	00	12	FE	23	7E	00	20	05	21	1B	12	3C	13	32	06	1B9
9180	00	CD	40	11	3B	D9	CD	49	12	3E	02	32	20	00	70	CF	155
9190	E1	D1	C1	C9	C5	04	1B	02	3E	03	32	20	00	13	1B	BF	11F
91A0	C5	06	0B	1A	0E	2B	09	23	23	10	F8	17	13	C1	C9	121	
91B0	7B	32	07	00	23	B5	E5	23	56	EB	7E	B5	2B	0A	7A	2D	135
91C0	00	3D	2B	36	3D	2B	01	29	22	2E	00	D1	17	1A	47	E6	1A5
91D0	F0	FE	30	2B	05	3A	05	00	1B	07	13	7B	E6	0F	32	05	160
91E0	00	4F	06	00	21	30	12	09	D5	5E	5A	1D	07	42	62	144	
91F0	6A	19	10	FD	D1	C1	44	40	AF	C9	CB	3C	1B	1D	1B	CB	1FA
Sum	24	0E	A9	5B	DF	A3	1A	9C	CA	D4	F2	F5	70	50	D9	29	1BF

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
9200	43	25	01	44	05	01	45	E9	00	46	DC	00	47	C3	00	41	14E
9210	AE	00	42	9B	00	52	00	60	43	15	01	44	F6	00	45	DC	191
9220	00	46	CF	00	47	BB	00	41	A4	00	42	92	00	52	00	00	11F
9230	01	02	03	04	06	08	0C	10	1B	20	0B	0F	0D	0C	0B	0A	1B1
9240	09	0B	10	0E	0D	0B	0B	0A	0B	0C	04	0B	0C	95	0B	0B	14D
9250	D5	C5	C5	21	39	12	3A	06	00	B7	2B	04	01	07	00	09	1FF
9260	3A	07	00	4F	3D	20	05	7E	02	32	20	00	09	46	7A	2D	147
9270	00	7D	2B	07	3D	2B	06	CB	3B	1B	02	CB	20	D1	21	00	1D1
9280	00	19	10	FD	44	40	2A	2E	00	7C	B5	2B	06	CD	39	11	1B5
9290	C1	D1	C9	E5	7E	04	32	3C	11	21	25	01	EF	13	7E	05	1BD
92A0	32	3C	11	E1	1B	EA	00	00	00	00	00	00	00	00	00	00	162
92B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100

92C0	B0	07	02	50	00	D0	0D	00	D6	77	00	1B	0A	00	B0	07	1AC
92D0	02	50	B0	07	02	50	00	D0	B0	07	B0	07	02	50	00	D0	12B
92E0	06	00	B0	3B	41	07	06	00	20	0B	3B	07	BB	3B	1F	D0	122
92F0	B0	2A	E5	0A	EE	01	BF	0A	B0	2A	B0	2A	7B	0B	00	13B	
Sum	05	25	E3	C4	DD	DB	CF	97	4B	96	3C	3B	65	47	BC	F2	1BB

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
9300	01	4D	6F	6E	69	74	6F	72	20	54	53	2D	31	30	30	30	19E
9310	00	0D	00	15	00	B0	00	00	00	00	00	00	00	00	00	00	1AF
9320	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9330	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9340	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9350	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9360	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9370	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9380	EF	1E	1E	10	DA	F0	13	EF	26	6C	6F	61	64	69	6E	67	1DC
9390	20	0B	0F	2C	EF	11	DA	F0	13	3E	14	32	D5	05	C3	00	139
93A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
93B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
93C0	0D	0B	0D	0B	0D	0B	0D	0B	0D	0B	0D	0B	0D	0B	0D	0B	1D0
93D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
93E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
93F0	3E	14	32	D5	05	7E	01	32	F0	00	C7	00	00	34	30	0D	1F7
Sum	95	B3	AE	BB	5E	5A	BA	AA	70	25	C4	E7	91	F9	BB	CB	1DC

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum	
9400	49	4E	54	20	50	52	49	4E	54	20	50	52	49	4E	54	20	165	
9410	50	52	49	4E	54	20	50	52	49	4E	54	20	50	52	49	4E	193	
9420	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0	
9430	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
9440	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
9450	00	00	00	00	00	00	4F	37	CF	D7	44	20	12	0F	44	00	1F5	
9460	52	55	4E	20	20	7F	0D	00	00	00	00	00	00	00	00	00	1C1	
9470	4C	49	53	54	20	7F	0D	00	00	00	00	00	00	00	00	00	1E9	
9480	53	59	53	54	45	4D	20	7F	0D	00	00	00	00	00	00	00	191	
9490	4D	4F	4E	20	20	7F	0D	00	00	00	00	00	00	00	00	00	1B6	
94A0	50	52	49	4E	54	20	00	00	00	00	00	00	00	00	00	00	1BA	
94B0	52	45	54	55	52	4E	20	00	00	00	00	00	00	00	00	00	1D0	
94C0	42	7F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1CE	
94D0	47	52	41	50	4B	20	00	00	00	00	00	00	00	00	00	00	19F	
94E0	53	41	56	45	22	0D	00	00	00	00	00	00	00	00	00	00	16B	
94F0	4C	4F	41	44	20	7F	0D	00	00	00	34	0D	52	4C	43	33	0D	12E
Sum	A0	DD	60	D1	7B	55	75	62	74	7B	F4	E3	F7	F2	14	BB	19A	

使い方は簡単に、S-ADRでスタート・アドレス、E-ADRでエンド・アドレスを入力してください。画面1でしか見られませんが、スペース・バーを押している間、画面は静止し、ブレーク・キーで中断されます。

2080	CD	2E	0A	CD	7B	05	CD	83	1A1
2090	05	EB	2B	2B	06	CD	BC	05	1D3
20A0	06	0B	0E	06	CD	C4	0B	7E	133
20B0	F5	B1	4F	F1	CD	D0	05	E5	14A
20C0	AF	ED	52	E1	2B	1C	CD	62	142
20D0	05	2B	17	0B	EA	FE	FD	20	12C

2080	FA	23	1B	0B	CD	C4	0B	7E	1E4
2090	3A	CD	C6	0B	79	CD	D0	05	1FD
20A0	1B	CB	CD	C4	0B	7E	3A	CD	1C1
20B0	C6	0B	79	CD	D0	05	C3	B1	16A
20C0	0B	0B	0B	0B	0B	0B	0B	0B	100



MZ-80B

WICS 303

2 インタープリタ

キャリア・ラボ 佐々木哲哉/山本耕司

現在ではほとんどのマイクロコンピュータにBASIC 言語が備わっており、数値計算には非常に便利なのですが、一般的に言ってBASICは制御プログラムやグラフィックス処理などの高速さを要求される仕事には向きません。

通常ならば、そのような仕事に対しては、アセンブラなどでプログラムを開発していたのですが、プログラムの内容が複雑になるにつれてアセンブラによる開発は困難になってきます。

そのような用途のために、私たちは速度とメモリ効率の点で、アセンブラ (=機械語) にひけをとらないくらいに効率のよいZ80コンパイラと、それと同じ文法のインタープリタとの複合システム (WICS) を開発しました。

なぜWICSなのか

普通、私たちがコンピュータに何か複雑な仕事をやらせる場合、コンピュータにプログラムを与えることが必要になります。

ところで、コンピュータが直接実行してくれるプログラムは、必ずそのコンピュータの機械語で書かれているものだけです。その機械語というのは、人間にはほとんど理解し難い数字 (十進法) の羅列にしか過ぎないものです。

これでは、どうにもならないので、その機械語の意味が少しは連想できそうな適当な単語 (ニモニックと呼ぶ) に置き替えたものがアセンブリ言語です。アセンブリ言語 (アセンブラとも言う) は、コンピュータのハードウェア (CPU) に密着したもので、CPUが異なれば当然異なり、普遍的なものではありません。

そこで、もっと人間にとって理解しやすい言語でプログラムを書いて、それを機械語に変換させよう、というのがコンパイラ (翻訳系) の考え方です。

ところで、BASICという言語が普及していますが、BASICはほとんどがコンパイラではなく、インタープリタ (解釈実行系) です。BASICは最近流行のPASCALのような構造言語から見れば、幼稚な言語ですが、その母体であるFORTRANとも似ているし、初心者にもわかりやすい言語です。

インタープリタはプログラムの作成や実行、デバッグが非常にやり易いのですが、その反面、プログラムの実行速

度が機械語に比べはるかに遅くなるという欠点があります。それもそのはず、インタープリタはその名のごとく、人間の書いたプログラムを毎回読みながらそれを解釈して実行しているからです。

では、BASICインタープリタで書かれたプログラムをコンパイラにかけて機械語にしてみればどうか、プログラムの作成とデバッグはインタープリタで行ない、完成したプログラムをコンパイルすれば、プログラムの開発は非常に楽になります。

ところで、マイコンではBASICのコンパイラはあまり一般には使われていません。実数系まで扱うとなるとコンパイラ自体がかなり巨大になり、テキストやオブジェクトを補助記憶に出入し入れしなければならなくなる、というのも理由の一つですが、8ビットCPUにとって、実数演算は苦手でかなり処理時間がかかるため、コンパイルできたとしてもあまり画期的な速度の向上は望めないからです。

また、マイコンを数値計算に使うだけであれば、インタープリタがいくら遅いと言ってもプログラム電卓よりは、はるかに速いし、数値計算の場合は、計算が終わるまで人間の方が待つていればよいのです。

ところが、マイコンというのは、もっと応用分野の広いものであり、速度を要求される制御関係やリアルタイム・ジョブ (TVゲームのことかな?) に使おうとすれば、処理速度が遅くては、使いものにならないのです。

そこで、BASICの使いやすさは捨てずに、(インタープリタ・モードを持たせる) アセンブラにひけをとらないくらいのコンパイラを実現しようとしたのが、このWICSなのです。

開発にあたって

言語の開発では、まず文法を決めるのが先ですが、大まかに言えば、WICSはBASIC文法にハードウェア制御用やメモリ操作に便利なコマンドなどを付け加え、その代り実数演算機能は省いています。さらに、コンパイラを記述しやすいように再帰呼び出し可能にして、コンパイラをWICS自身で記述することとしました。また、インタープリタは自己記述は少し無理なので、BASEアセンブラで記述しています。

システムの概略

WICSは大きく分けて、インタープリタ、コンパイラ、マシン語モニタを含んだランタイム・パッケージの3つの部分に分かれています。ランタイム・パッケージはメモリ常駐になっており、インタープリタとコンパイラ、それにユーザーの作成したコンパイル・オブジェクトはすべてこのパッケージに依存します。

WICSでのプログラム開発形態は、かなりユーザーの自由になっていますが、次のようなものが考えられます。

I) インタープリタのみによる開発。

WICSインタープリタ(もちろんランタイム・パッケージを含む)のみを用いてプログラムを作る場合です。これは通常のBASICインタープリタを使うのとはほとんど同じ感覚です。実数演算はありませんが、実数型のBASICに比べて2~3倍の速さを持っています。また、WICSテキスト・プログラムはメモリが許せば、いくつでも置くことができます(TEXT命令で切り換えて使用)。

II) インタープリタとコンパイラを両方使う場合。

インタープリタとコンパイラはメモリ上に共存できるので、両方ともメモリにロードしておいて、インタープリタでプログラムを編集、デバッグして、即、コンパイラにかけることができます。

また、コンパイルされたサブルーチンは、インタープリタからコールすることも自由にできます。このとき、サブルーチン内の変数名とインタープリタで使う変数名が重複しても問題はありません。

サブルーチンへの引数渡しはPUSH, POP命令で行なうか、あるいはメモリのフリーエリアに共通の配列をとって行なえます。

マシン語とのリンクも自由にできます(ただし、インデックス・レジスタはIX, IYともにソフトウェア・スタックなので、破壊してはいけません)。

III) インタープリタを切り離す場合。

ユーザープログラムが大きくなって、ユーザープログラムのオブジェクトを作る場所がなくなったときは、インタープリタを切り離して、インタープリタ本体のメモリ領域を、オブジェクト生成に使うこともできます。

また、インタープリタ本体の代わりに、そこにユーザーオブジェクトを置き、ランタイム・パッケージと共にセーブすれば、完全独立型オブジェクトを作ることもできます。いずれの場合もインタープリタを切り離すとテキスト・プログラムの編集はできなくなります。

特筆すべきことは、コンパイラがかなりのレベルまでオブジェクトの最適化(生成コードの縮小と高速化)を行なうため、ソース対オブジェクト比が1対0.6~0.8程度にまで縮まり、実行速度は1桁以上向上することです。

また、コンパイル速度は、約400行/分と高速です。

文法上の特徴

WICSの文法はBASICとかなり似ています。なぜ、BASICの構造を選んだかという問いに対しては、BASICの文法が広く普及している、ということのほかに、BASICの制御構造が、機械語で比較的实现しやすい、ということと言えます。

このことは、生成されるオブジェクトの効率を良くすることにもなります。もしも、PASCALなどの言語構造を使うと、ローカル変数の割り当てや手続き呼び出しの引機構造のために、かなりのオーバーヘッドが生じてしまいます。

ただし、BASIC文法と異なる部分も多少あります。以下の、

① 変数名は英字(アルファベット)で始まる英数字(～Zと0～9)からなる任意長で、予約語(FORやGOSUBなど)で始まるものでなければ、すべての文字を識別します。すなわち、"HENSU"という名前の変数と、"HENSU1"という名前の変数は区別して扱われます。

② 配列はBASICの配列と異なっていて、

ユーザーがメモリのフリーエリアに指定する形式で、1バイト配列と2バイト配列があります。いずれも配列の要素はメモリ番地と直接対応しているため、メモリ操作ポイントと使うこともできます(機械語におけるインデックス・アドレッシングの考え方に似ています)。

③ ダブル・クォーテーションで囲まれた1文字は、文字定数として、整数値(そのASCIIコード)に変換されます。

④ 演算子にOR, AND, XORがあり、これによりビット操作を可能にします。

⑤ その他、再帰呼び出しを可能にするPUSH, POP命令があり、FOR～NEXTやGOSUB～RETURNとは別の独立のスタックを持たせているので、引数をPUSHしたあとでサブルーチンをコールする方法が使えます。

(文法上の注意)

- ① GOTO, GOSUBの飛び先番号は必ず定数でなくてはいけません。
- ② ただし、マシン語を呼び出すCALLでは、番地の代りに式や名前(変数)も使えます。
- ③ FORに対するNEXTはコンパイラにかけられることを考えて、必ず1対1で対応させること。ネスティングは可能(スタックの許す限り)ですが、途中からの飛び出しや飛び込みはコンパイラで走らせる場合、暴走の原因、非構造化の始まりになります。

(WICSの扱える数値および変数)

① 変数

- 変数は英字(A～Z)で始まり、英数字(A～Z, 0～9)からなる任意の文字列です。
- 変数名の長さに制限はありません。また、変数名は全文字を識別します。
- ただし、予約語(FORやGOSUBなど)で始まる文字列は変数とはみなしません。予約語が変数名の途中から現われる場合はありません(変数名とみなす)。
- 変数は-32,768～32,767までの整数値をとることができます。

② 配列 (INDEX)

- 配列は、通常のBASICのDIM文と異なり、ユーザーがメモリのフリーエリア上に指定するようになっています。
- 2バイト配列: 変数名(式)
2バイト配列は(変数の内容+式の値*2)のメモリ番地の2バイト分の値に対応します。
- 1バイト配列: 変数名(式)
1バイト配列は、(変数の内容+式の値)のメモリ番地の1バイトの値に対応します。

WICSでは配列(INDEX)というものは機械語のインデックス・アドレッシングに相当するもので、直接的にメモリを操作できます。

③ 定数

- i) 10進定数: -32768～32767の範囲の整数。
- ii) 16進定数: \$0000～\$FFFF (16進数は"\$"をはじめに付けると)。
- iii) 文字定数: "A"のように、クォーテーション" "でくくられた1文字のASCIIコード(整数値に変換される。)

④ 式

- 式: = <項> [(<演算子> <項>)]
式は、項と2項演算子より成り立っています。
- 2項演算子を、優先順位の低い順に示します。

- 1) 不等号, 等号: >, =, >=, <=, =, <, <=, <, >
 2) 加減算: +, -, OR
 3) 乗除算: *, /, AND, XOR
 ●() を含む場合は, () の中が最優先されます。
 ●等号, 不等号演算子は, 左項<演算子>右項が成立すれば, その

値は1(真)で不成立の場合は, 0(偽)の値をとります。

- OR, AND, XORは, 左項と右項の値を16ビットの2進数値とみなして, 各ビットごとに, OR, AND, XORをそれぞれ行ない, 16ビットにまとめて値にするものです。

予約されているコマンド, ステートメント, ファンクション名

(コマンド)

MON	●モニタに制御を移します。	NEW	前とその値をすべて表示します。	APPEND ("ファイル名")	●カセットに記録された"WICS"テキストを現在のテキスト・エリア1のプログラムの後につづけてロードします。
BYE	●MZ-80Bのモニタに戻ります。	LOAD ("ファイル名")	●プログラムをすべて消去します。	●APPENDでは行番号はそのままですので注意してください。	
RUN (行番号)	●インタプリタ・モードでプログラムを実行します。	●カセット・テープからテキスト・プログラムを現在のテキスト・エリアにロードします。	●ファイル名を省略すれば, 最初にあったプログラムをロードします。	SYSTEM	●現在のテキスト・エリアの先頭, 終り, 1回の番地をそれぞれ表示します。
LIST (行番号) [,] (行番号)	●行番号を省略すると変数テーブルをクリアして, プログラムを初めから実行します。	SAVE ("ファイル名")	●LOADの逆でテキスト・エリアにある"WICS"のテキスト・プログラムをカセットにセーブします。	LIMIT	●テキスト・エリアの1回を定義します。
DELETE (行番号), (行番号)	●テキスト・プログラムを表示します。	VERIFY ("ファイル名")	●カセットに記録された"WICS"テキストをテキスト・エリアにあるプログラムと比較します。	TEXT (番地)	●通常はC000番地になっています。
VLIST	●リスト行を何行目から何行目まで, という指定もできます。	●ファイル名を省略すると最初に見つけたプログラムとの比較をします。		CLR	●テキスト・エリアの先頭番地を設定します。
	●指定された範囲の行を削除します。				●同時に終り番地は自動セーブされます。
	●インタプリタ・モードで使われた変数の名				●インタプリタで使った変数をクリアします。

■以上がインタプリタのコマンドです

これらはダイレクト・モード, テキスト・プログラム中いずれでも使えますが, コンパイルは無理なので注意してください。

(ステートメント)

IF (式) THEN (文)	●式の値が真 (1) 以外の値はすべて真とみなすのとき THEN 以下の文を実行します。 ●THEN は省略できます。	1) 1の形式は, " (コンマ) で区切れば, いくつでも重ねることが出来ます。 注: 変数ベースは, 次のようになります。 PRINT %5, 100 5文字 注: PRINT 文は, 最後に改行を行いませんので, 改行したいときのみ, " (スラッシュ) を入れること。 注: % (10進出力のフォーマット) は, 次の"%" が来るまで有効です。	指定されるアドレスを行パツファとして, そのアドレスに1行分の文字を格納します。また, 文字列の最後には終りコードとして \$ 0 D を格納します。
FOR (変数) = (初期値) TO (最終値) [STEP (変化幅)] [繰り返すプログラム]	●BASICと同じ繰り返し命令です。 ●STEPが1のときは, STEP文は省略できます。 ●ただし, STEP値は正の値でなければなりません。	INPUT (変数名) [, (変数名) ...]	STA (式), (変数)
REPEAT [繰り返すプログラム]	●式の値が真になるまで, REPEATとUNTIL にはさまれたプログラムを繰り返して実行します。	●キーボードから数字を入力して, 変数に代入していきます。数字は10進, および16進(\$) をはじめに人力) が可能です。 ●"WICS"では文字列に対してBASICとは違う取り扱いをしているので, 変数変数がありません。したがって, INPUT AS などはできないので注意してください。 (このような場合には, LINPUTを使います。)	●CPUがアクセスできるRAMをグラフィックRAM1に設定します。
UNTIL (式)	●式の値が1であれば行番号1へ, 2であれば行番号2へというように式の値の値にある行番号へ飛びます。ただし, その値に該当する行番号が書かれていないときは, 分岐せずに次の命令へ行きます。	STOP	●式2: CPUがアクセスできるRAMをグラフィックRAM2に設定します。
ON (式) GOTO (行番号1), (行番号2), ..., (行番号N)	●式の値が1であれば行番号1へ, 2であれば行番号2へというように式の値の値にある行番号へ飛びます。ただし, その値に該当する行番号が書かれていないときは, 分岐せずに次の命令へ行きます。	●プログラムの実行を中止して, 中止された行番号を表示します。	●式3: CPUがアクセスできるRAMをグラフィックRAM1に設定します。
ON (式) GOSUB 前項と同じ	●ON-GOTOと同じように式の値の番地にある行番号をサブルーチン・コールします。	END	●式4: CPUがアクセスできるRAMをグラフィックRAM2に設定します。
GOTO (行番号)	●行番号に分岐します。	CALL (式)	●式5: CPUがアクセスできるRAMを白でクリアします。
GOSUB (行番号)	●行番号をサブルーチン・コールします。	●式の値の番地にあるマシン語サブルーチンをコールします。	●式6: CPUがアクセスできるRAMを白でクリアします。
RETURN	●サブルーチンの呼び出しを完了します。	PUSH (式) [, (式) ...]	●式7: CPUがアクセスできるRAMを白でクリアします。
REM	●RETURNを実行すると, GOSUBでコールしたプログラム (サブルーチンの呼び出し) のGOSUB文の次へ戻ります。	●式の値を順次, スタックに積み込みます。 POP (変数名) [, (式) ...] ●スタックに積み込まれた値をおりし, 順次, 変数に代入します。	●式8: CPUがアクセスできるRAMを白でクリアします。
PRINT	●変数や数値を出力装置に出力します。出力装置は, 通常は, ディスプレイが指定されます。 ●形式は, 次のようなものがあります。 ① / (スラッシュ): 改行を行なう。 ② 文字列: 文字列を出力。 ③ 式: 式の値を定められた変数, スペースで右詰め出力する。 ④ # 2 式: 式の値を16進2桁で出力する。 ⑤ # 4 式: 式の値を16進4桁で出力する。 ⑥ & 式: 式の値のASCII文字を出力する(1文字)。 ⑦ / 式: 式で定められた番地に格納されている文字列を \$ 0 D まで出力する。 ⑧ % 式: 変数ベースを式の値に定める。 ⑨ + 式: 式の値を 0-65535までの10進数で (負の値をしないで) 出力する。	OUT (I/O 番地), (式)	●式9: CPUがアクセスできるRAMを白でクリアします。
		●I/O 番地に式の値を出力します。 ●どちらも0-255までの値です。	●式10: CPUがアクセスできるRAMを白でクリアします。
		CURSOR (式1), (式2)	●式11: CPUがアクセスできるRAMを白でクリアします。
		●CRTのカーソルX, Yの座標を式1と式2で指定します。	●式12: CPUがアクセスできるRAMを白でクリアします。
		POKE (式1), (式2)	●式13: CPUがアクセスできるRAMを白でクリアします。
		●式1のアドレスに式2の値の下位バイトが代入されます。	●式14: CPUがアクセスできるRAMを白でクリアします。
		CURR (式)	●式15: CPUがアクセスできるRAMを白でクリアします。
		●式の値をカーソルのX座標とします。	●式16: CPUがアクセスできるRAMを白でクリアします。
		CURV (式)	●式17: CPUがアクセスできるRAMを白でクリアします。
		●式の値をカーソルのY座標とします。	●式18: CPUがアクセスできるRAMを白でクリアします。
		PLOT (式X), (式Y), (式M)	●式19: CPUがアクセスできるRAMを白でクリアします。
		●式XはXの座標(0-79), 式YはYの座標(0-49)とし, 式MのMODEで, ドットをプロットします。	●式20: CPUがアクセスできるRAMを白でクリアします。
		MODE = 0 ドットをリセットします。 MODE = 1 ドットをセットします。 MODE = 2 ドットをリバーズ(反転)します。 MODE = 3 ドットをリバーズ(反転)します。	●式21: CPUがアクセスできるRAMを白でクリアします。
		LINPUT (式)	●式22: CPUがアクセスできるRAMを白でクリアします。
		●1行人力を行ない, 入力された行の文字を式で	●式23: CPUがアクセスできるRAMを白でクリアします。

注意：格納の順序は、次のようになります。また、 $(X-1)$ の値は8の倍数でなければなりません。

(X,Y)	ビット0		ビット7	
	ARRAY(0)	ARRAY(1)	ARRAY(2)	ARRAY(3)
	ARRAY(4)	ARRAY(5)		

(X1, Y1)

PAINT (X<X>, Y<Y>)

- グラフィック RAMの座標 (X, Y) の点を含む閉空間を塗りつぶします。
- 閉空間の図形が重複する場合は、IVスタックがオーバーフローすることがあります。

<ファンクション>

GET

- キー入力が行なわれるまでカーソルを点滅し、キー入力を行なったときそのASCIIコードを値とします。

INKEY

- リアルタイム入力です。キーが押されていなければ0、押されていればそのASCIIコードを値とします。

MODE (X<X>, Y<Y>)

- 式1を式2で割った余りを値とします。

ABS (X<X>)

- 式が負であれば正の数にします。正であれば、それを値とします。

SGN (X<X>)

- 式の符号を値とします。
式が負であれば-1
式が0であれば0
式が正であれば1を値とします。

RND (X<X>)

- 1-式までの乱数を値とします。

INP (I/O 基地)

- I/O 基地の内容を値とします。

ADRS (X<変数>)

- 変数の格納されているアドレスを値とします。

PEEK (X<X>)

- 式のアドレスのメモリ内容を値とします。

INC (X<変数>)

- 変数の値を1つ増加させ、その値を値とします。

DEC (X<変数>)

- 変数の値を1つ減少させ、その値を値とします。

CIRCLE (X<X>, Y<Y>, R<R> [, C<CM>])

- 中心 (X, Y) 半径Rの円を描きます。X, Y, Rは画面をはみ出した場合、画面に映る部分のみ描きます。

BELL

- ベルを鳴らす。

MUSIC (音楽データ)

- 音楽データを演奏します (BASIC SP-5030 の仕様と同じ)。

SOUND (X<X>, Y<Y>)

- 式1の高さの音と式2の長さだけ鳴らします。
- 音の高さと長さは、音階 (ドレミ) とは異なり、パルスのカウント値とパルス数に対応しています。

USR (X<X>, Y<Y>)

- 式2をHレジスタに代入した後に、式1のアドレスをファンクション・コールします。
- RETURN時のHレジスタを値とします。

SPEEK (X<X>, Y<Y>)

- CRT 1の (X, Y) 座標にあるキャラクタのディスプレイ・コードを値とします。

CP# (X<X>, Y<Y>)

- 式1を何回繰り返す (X, Y) はそれぞれ2桁の式1の先頭アドレスを指します。CP3は比較の最終コードを指し、式1が指す式1何回に、このコードが現れたとき比較を終わります。一致のときは1、不一致のときは0を値とします。

CPI (X<X>, Y<Y>)

- 式1の値を先頭アドレスとする式1何回に、ファンクション内の式1何回に一致します。一致のときは1、不一致のときは0を値とします。

TIME 関数

- 内部のタイマの値を値とします。タイマの値をセットする場合は、TIME=式とします。TIMEの単位は秒です。BASICのように60進数にフォーマットされません。

DOT (X<X>, Y<Y>)

- グラフィック RAMの座標 (X, Y) に点が打たれていれば値は1、なければ0になります。
- (X, Y) が画面をはみ出した場合、1を返します。



この長さを式1で決める

パルスの数が式2で決まる

TEMPO (X<X>)

- MUSICの演奏スピードを指定します。

PRMODE (X<X>)

- PRINT出力の機器を指定します。
0: CRTのみ出力。
1: PRINTERのみ出力。
2: CRT, PRINTER同時出力。

メモリー・マップ



WICSインタープリタ ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
1500 C3 F3 16 C3 B4 2B C3 A5 1B C3 04 17 C3 07 13 C3 17
1510 0B 17 C3 10 17 C3 D4 1C C3 D8 1C C3 35 17 C3 3E 16A
1520 17 C3 4B 17 C3 A2 17 C3 A6 17 C3 AC 17 C3 92 19 129
1530 C3 10 C3 38 1B C3 67 1A C3 A9 1A C3 29 1C C3 152
1540 1F C3 C3 0E 1C C3 12 1C C3 FA 1B C3 EC 1B C3 17 192
1550 1C C3 C3 1B C3 0F 1B C3 20 1B C3 39 1B C3 13 1B 1C3
1560 C3 F5 1C C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 1A3
1570 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 1CF
1580 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 1CF
1590 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 00 C3 192
15A0 00 00 00 E0 05 00 D2 04 00 00 00 00 00 00 00 1B8
15B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
15C0 7C A2 67 7B A3 C9 67 C7 B2 67 7B B3 6F C9 7C AA 100
15D0 67 7B B3 C9 67 C7 B2 67 7B B3 6F C9 7C AA 100
15E0 52 21 00 00 C8 2C C9 13 B7 ED 52 21 00 00 F0 2C 176
15F0 C9 13 B7 ED 52 21 00 00 FB 2C C9 7C FE 80 DB 7C 12E

Sum1 67 07 6C 15 F3 B0 75 72 D3 90 C5 6F F8 8D 1C D4 135

```

```

1700 24 EF 17 C9 EF 09 C9 AA 15 85 87 FA 23 17 F5 EF 09 F1 3D 109
1720 C3 17 17 EB CD 4A 1C EB C9 CD FF 15 CD A2 1C 2B 15A
1730 36 2D C3 13 17 7E 23 FE CD 0B EF C3 1B 7F E3 7E 126
1740 23 C3 E7 C8 EF C3 08 F6 3A 03 00 F5 CD 44 1C F1 1D5
1750 B3 5F CD ED 1C EF CD 0A 80 1B 0F FE 24 2B 0E 1B9
1760 2D 20 07 13 CD 92 1C C3 FF 15 C3 92 1C 13 1A CD 124
1770 92 17 DA 6C 1B 1B 0B 13 1A CD 92 17 DA 2A 2A 1EB
1780 00 6F 13 1A CD 92 17 2B 29 29 29 29 06 04 4F 09 1EC
1790 1B 0F D6 30 DA 0B FE 0A 3F CD D6 07 FE 0A D6 FE 10 1CB
17A0 3F C9 EF 0A D6 0E 0E CD 26 06 6F C9 CD E6 17 DB 116
17B0 CD 31 1B 3A AB 15 B7 2B 0C EF 01 2B 04 79 AE 1B 162
17C0 07 7D B3 80 13 79 2F A6 77 DB E8 E4 7F D3 E8 09 1C2
17D0 CD E6 17 30 04 21 01 00 C9 CD 31 1B 79 A6 21 00 13F
17E0 00 2B 01 2C 1B ED 2A AC 15 7D E6 07 47 CB 3C 1C5
17F0 CB 1D 7C 2B 30 C8 CB 3D CB 3D 7D FE 2B 30 27 E5 15A

Sum1 12 AB CA 0E 0E DA 6D 0B 74 B2 F0 1F 7A 16 94 D2 19C

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
1600 2F 67 70 2F 6F 23 C9 7C B5 0B 7C 21 01 06 F5 80 1B2
1610 DB 28 2B C9 FD 2B FD 70 0D FD 2B FD 75 0D FD 7C 1A3
1620 FE 20 D0 C3 6F 1B FD 6E 0D FD 23 FD 66 0D FD 23 149
1630 C9 06 7A B7 20 06 B3 20 C3 9A 1B FD 46 16 1CB
1640 04 EB CD FF 15 EB 7C 06 3B 04 05 C8 F2 15 7E 14F
1650 B7 2B 06 CD 5D 1B 18 47 00 44 20 04 21 00 3E 0B 1D8
1660 CB 21 CB 10 ED 6A 0C ED 32 02 19 0D C8 21 CB 17B
1670 10 ED 0A CD ED 32 02 19 0D C8 21 CB 17B
1680 EB 09 CD 31 16 0E C9 44 21 00 00 7A 7B C4 92 1B5
1690 16 7B 29 37 10 01 09 29 17 30 01 09 29 17 30 01 1F4
16A0 09 29 17 30 01 09 29 17 30 01 09 29 17 30 01 09 177
16B0 29 17 30 01 09 29 17 30 01 09 29 17 30 01 09 198
16C0 09 3D CD 87 16 2B 22 A6 15 D1 7A 93 20 04 ED 53 11A
16D0 A6 15 CD 87 16 2B 22 A6 15 D1 7A 93 20 04 ED 53 11A
16E0 00 3E 10 29 EB ED 6B 30 04 09 30 01 13 30 C2 124
16F0 E3 16 C9 AF 32 36 00 31 FC 1F FD 21 FE 23 30 00 127

Sum1 29 03 30 D1 73 9D 00 1B B7 DA 93 9A 7B FD 1A 9B 137

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
1800 2A AE 15 7C B7 20 1E 7D FE CB 30 19 54 50 29 1ED
1810 19 29 29 29 EB 21 29 1B 7B B5 6F AE E1 19 ED 5B 1D8
1820 A2 15 19 D1 C9 E1 D1 37 C9 01 02 04 0B 10 20 49 19B
1830 80 DB 09 F6 80 D3 EB C9 CD E6 17 DB 59 09 ED 5B 159
1840 80 D3 21 C0 FE 19 DB 2A AC 15 EB 87 ED 52 30 22 1B3
1850 19 ED 53 80 15 22 AC 15 EB 87 ED 52 30 22 1B3
1860 ED 5B 82 15 ED 53 AE 15 22 82 15 ED 53 82 15 ED 1F7
1870 E1 DB 22 1B 15 ED 53 AE 15 22 82 15 ED 53 82 15 ED 1F7
1880 E1 C9 2A AE 15 22 82 15 ED 53 AE 15 22 82 15 ED 1F7
1890 87 ED 52 AE 15 22 82 15 ED 53 AE 15 22 82 15 ED 1F7
1900 85 19 ED 4B 8B 15 7B 87 2B 21 60 69 03 CD FF 15 142
1910 22 D6 1B 2A BA 15 22 12 1D CD 65 19 ED 53 1B 180
1920 21 00 00 CD 31 1B D9 7B 00 00 77 CB 03 30 01 23 124
1930 D9 11 00 00 09 11 00 00 09 19 3B 03 1B ED 03 09 132
1940 09 D9 0B 7B 81 C2 CE 1B C3 C9 17 79 09 57 2B 1ED
1950 CD 65 19 ED 53 02 19 3A BA 15 47 CD 31 1B AF D9 194

Sum1 09 F0 2C AD F7 F0 73 F4 2B 87 C3 3A 1F 0D 4A C4 106

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
1900 0B 7B 00 00 77 CB 03 30 01 23 D9 0B 80 3B 02 B9 170

```

1910	3F	38	08	D9	15	C2	00	19	C3	C9	17	91	D9	09	15	C2	i35
1920	00	19	C3	C9	17	3A	BA	15	4F	D9	57	14	D9	08	B3	19	i9A
1930	CD	65	19	ED	53	42	19	3A	BB	15	47	CD	31	18	4F	D9	iD2
1940	08	78	00	00	77	09	08	80	3D	02	B9	3F	3B	09	19	4F	iAF
1950	15	C2	40	19	C3	C9	17	91	D9	09	15	C2	03	11	23	12	i36
1960	40	19	C3	C9	17	3A	BA	15	4F	D9	57	14	2F	A6	C9	FE	i7B
1970	01	20	04	11	B6	00	C9	FE	02	20	04	11	AE	00	C9	11	i72
1980	00	00	00	C9	D9	08	01	28	00	FE	01	20	03	01	DB	FF	iD5
1990	D9	C7	2A	84	15	44	4D	CD	FF	15	11	00	00	CD	BB	19	iB9
19A0	13	19	19	28	7C	FE	B0	30	07	08	B7	ED	42	ED	42	78	iAF
19B0	E5	60	49	B7	ED	52	E1	F2	90	19	CD	C9	19	CD	C4	i36	
19C0	19	CD	C9	19	C5	D1	C1	D1	C9	CD	EE	19	CD	E6	19	CD	i2A
19D0	EE	19	CD	DE	19	CD	EE	19	CD	EE	19	CD	EE	19	78	2F	iE6
19E0	47	79	2F	4F	03	C9	7A	2F	57	7B	2F	5F	13	C9	03	2A	iFE
19F0	B0	15	09	22	AC	15	2A	B2	15	19	22	AE	15	C9	02	1A	iB9

Sum	41	5D	2E	39	10	2A	6D	EE	B0	46	35	4F	31	E4	7E	19	

1E10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1E90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1EA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1EB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1EC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1ED0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1EE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
1EF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00

Sum1	41	5D	2E	59	10	2A	60	FE	B0	9E	A4	35	6F	31	E4	7E	iB6
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1A00	E1	C9	D5	D5	E5	CD	AC	17	E1	B1	C1	C9	3A	AE	15	4F	i61
1A10	26	00	29	29	29	44	40	29	29	09	3A	13	00	FE	4F	20	i87
1A20	01	29	3A	AC	15	4F	06	D0	09	09	2A	AC	15	11	00	FE	iD6
1A30	19	08	2A	B0	15	11	00	FE	EB	19	D8	2A	AC	15	EB	87	i18
1A40	ED	52	08	CD	3C	B8	1D	CD	30	CB	30	23	22	08	1A	B8	iB9
1A50	AE	15	FE	CB	3F	ED	52	08	CD	3C	B8	1D	CD	30	23	22	i5A
1A60	3C	32	00	CD	3C	B8	1D	CD	30	23	22	08	1A	B8	1D	CD	i5E
1A70	84	15	3A	00	18	47	D9	E5	D9	2A	08	1B	CD	31	18	D9	i48
1A80	5E	23	56	CD	9D	1A	CD	C9	17	7B	D9	12	13	2D	C2	7C	iEC
1A90	1A	09	D5	11	28	00	19	D1	D9	10	DA	C9	7B	87	CB	4F	iAF
1AA0	CB	1A	CD	1A	CD	1A	CD	1A	CD	1A	CD	1A	CD	1A	CD	1A	i11
1AB0	65	19	ED	53	F3	1D	3A	BA	15	4F	D9	57	14	2F	A6	C9	i37
1AC0	E5	60	49	2A	08	1B	AF	08	1A	CD	1A	2D	2B	09	1A	13	i3B
1AD0	08	1A	CD	EA	1A	1B	F4	1A	13	0B	AF	08	EA	1A	13	0B	i6E
1AE0	D5	11	28	00	19	D1	D9	10	D6	C9	CD	FC	1A	08	CD	11	i11
1AF0	31	18	08	00	00	77	23	CD	C9	17	D9	C9	04	05	CB	08	i13

Sum1	7A	21	AE	1E	D1	6F	D0	26	F8	22	C5	00	49	55	78	70	i28
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2000	51	51	51	00	41	44	52	00	45	00	57	57	00	49	00	00	i5A
2010	54	00	00	54	54	00	00	42	00	59	00	00	00	00	00	00	i9E
2020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
2090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00
20F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	i00

Sum1	A5	5E	A5	4E	75	31	93	1A	B7	1A	D0	64	57	0D	49	0D	1F
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum2
2100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
2190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
21F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100

Sum1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2260	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
22A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
22B0	00	00	00	00	00	00	00	00	4C	28	41	29	20	4C	3D	41	10
22C0	0D	20	4B	4C	28	30	52	45	54	0D	3B	0D	40	53	47	48	10
22D0	08	3F	08	40	28	32	20	52	55	54	3B	08	3B	3E	31	33	10
22E0	30	37	0D	20	30	48	20	54	4C	33	20	0D	20	41	33	30	10
22F0	20	41	3C	24	38	30	20	52	45	54	0D	20	48	4C	2D	2D	10

ダンプ・リスト

2310	29	3D	48	20	28	D9	49	59	29	3D	4C	20	20	20	20	17
2320	3B	31	33	31	34	00	20	21	28	46	44	29	20	41	3D	13
2330	20	49	46	20	41	3E	3D	49	59	42	54	40	48	20	52	45
2340	54	00	20	4A	40	50	20	53	50	4F	56	46	4C	57	00	38
2350	00	40	40	50	4F	50	4C	28	49	59	28	29	20	48	3D	1A8
2360	28	49	59	28	29	52	45	54	20	38	31	33	31	46	00	16C
2370	3B	00	44	49	56	53	20	42	3D	30	20	20	20	20	20	10D
2380	20	20	20	20	20	20	20	20	20	20	39	31	33	32	41	72
2390	00	20	41	3D	44	20	49	46	20	41	3C	30	32	20	47	4F
23A0	54	4F	20	49	46	56	30	00	20	41	3D	41	2E	4F	52	2E
23B0	45	20	49	46	20	3C	3E	20	47	4F	54	4F	4A	49	56	1E
23C0	31	00	20	47	4F	54	20	44	49	56	45	52	52	20	20	1B0
23D0	38	00	44	49	56	30	20	49	46	20	50	4C	55	53	20	47
23E0	4F	34	4F	20	44	49	56	31	00	20	42	2B	20	44	45	3C
23F0	3E	48	4C	20	21	43	00	31	81	D4	81	86	00	45	00	1D5
Sum: 27 11 DC 09 40 78 7C 94 72 FB DF BE 74 45 7E 6F 172																

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
2400	C3	00	19	EB	E1	C9	5E	BA	28	1A	FE	FF	08	13	13	ES
2410	00	19	EB	E1	C9	5E	BA	28	1A	FE	FF	08	13	13	ES	5E
2420	52	E1	30	06	ED	5B	80	1B	18	04	ED	5B	80	1A	1E	1B
2430	FF	20	02	87	0F	1A	13	8C	28	0C	30	13	13	83	5F	110
2440	74	CE	00	57	1E	EF	1A	18	28	02	1E	1E	13	1A	19	1E3
2450	87	C9	1A	FE	FF	08	13	8C	28	0C	30	13	13	83	5F	110
2460	24	2A	ED	80	1B	18	04	ED	5B	80	1A	1E	1B	1E	1B	1E3
2470	58	AA	10	EB	87	ED	52	44	00	D1	E1	C9	5E	BA	28	1A
2480	1A	FE	3A	CE	00	57	1E	EF	1A	18	28	02	1E	1E	13	1A
2490	6C	7C	05	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
24A0	CD	2A	2A	D5	ED	80	1B	18	04	ED	5B	80	1A	1E	1B	1E3
24B0	05	20	ED	80	1B	18	04	ED	5B	80	1A	1E	1B	1E	1B	1E3
24C0	10	21	34	2F	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E3
24D0	00	20	20	20	20	20	20	20	20	20	20	20	20	20	20	10D
24E0	0E	3B	18	2A	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E3
24F0	8B	28	F1	FE	20	47	13	7E	1B	1E	1B	1E	1B	1E	1B	1E3
Sum: 74 C1 C4 A7 F8 3F E5 AB DD 79 06 7C C3 80 49 6C 107																

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
2500	24	00	ED	5B	80	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
2510	00	ED	5B	80	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E3
2520	2E	33	30	5E	00	7F	00	20	23	13	ED	5B	80	1A	1E	1B
2530	DD	36	00	00	00	36	01	FF	DD	E1	D1	E1	C9	5E	BA	28
2540	80	DD	23	DD	71	00	00	23	1E	1E	1E	1E	1E	1E	1E	1B
2550	36	01	4A	FE	00	23	1A	13	DD	77	00	00	23	ED	53	42
2560	84	1A	FE	00	C2	57	23	1A	13	DD	77	00	00	23	ED	53
2570	23	ED	53	84	1A	FE	00	C2	57	23	1A	13	DD	77	00	00
2580	8B	24	1A	FE	00	77	00	00	23	1E	1E	1E	1E	1E	1E	1B
2590	53	84	1A	FE	00	77	00	00	23	1E	1E	1E	1E	1E	1E	1B
25A0	E5	25	E5	C5	2A	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
25B0	F4	2A	C1	E1	C5	7E	00	64	29	48	25	09	E1	D9	D1	DE
25C0	19	E5	C5	C1	D1	E5	3A	1A	1E	1B	1E	1B	1E	1B	1E	1B
25D0	23	71	E5	06	01	23	1A	0A	FE	00	20	04	77	01	70	1B0
25E0	C9	13	77	F1	05	01	04	10	1A	FE	00	20	02	D1	C9	117
25F0	03	13	77	F1	05	01	04	10	1A	FE	00	20	02	D1	C9	117
Sum: AB BE CD 1B 27 87 BE AE 83 13 14 AD 83 2E 87 B2 iFE																

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
2600	FE	00	28	0C	87	20	07	2A	80	1B	1E	1B	1E	1B	1E	1B
2610	23	18	EB	00	18	12	CD	22	28	31	DD	77	00	13	1E	1B
2620	26	1F	47	7E	23	FE	3A	C2	23	26	10	77	7E	23	DD	77
2630	00	00	23	FE	3A	C2	23	26	10	77	7E	23	DD	77	00	13
2640	DD	23	1A	13	DD	77	00	00	23	FE	22	28	31	DD	77	00
2650	C2	CD	56	26	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
2660	F1	3E	00	18	AE	05	EB	56	26	1E	1B	1E	1B	1E	1B	1E
2670	23	EB	00	C9	1A	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E
2680	EB	00	ED	80	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E
2690	1D	87	ED	42	08	CD	56	26	1E	1B	1E	1B	1E	1B	1E	1B
26A0	CA	B4	28	1D	87	ED	42	08	CD	56	26	1E	1B	1E	1B	1E
26B0	22	AC	10	28	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
26C0	18	12	CD	71	1C	22	AC	10	28	1E	1B	1E	1B	1E	1B	1E
26D0	AC	10	18	0E	13	CD	70	24	20	05	1F	FF	18	0E	20	1C
26E0	71	1C	1E	AE	10	D5	2A	AC	10	28	1E	1B	1E	1B	1E	1B
26F0	C9	2A	AE	10	7E	FF	20	04	22	AA	10	C9	23	23	23	172
Sum: 9C 85 AF 45 43 C7 7A 3B 80 97 CC 40 98 70 1E 169																

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
2700	7E	00	28	0C	87	20	07	2A	80	1B	1E	1B	1E	1B	1E	1B
2710	EE	23	18	EB	00	18	12	CD	22	28	31	DD	77	00	13	1E
2720	00	00	23	FE	3A	C2	23	26	10	77	7E	23	DD	77	00	13
2730	CD	28	05	2A	AC	10	18	12	CD	22	28	31	DD	77	00	13
2740	FF	10	03	CD	71	1C	22	AC	10	28	1E	1B	1E	1B	1E	1B
2750	42	DA	DD	2A	D5	2A	AC	10	28	1E	1B	1E	1B	1E	1B	1E
2760	2A	2A	ED	80	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E
2770	CD	D1	33	05	01	79	27	05	E9	D1	C9	2A	AE	10	7E	1B
2780	22	AA	10	21	00	38	01	00	04	36	20	28	78	B1	C2	196
2790	89	27	22	3C	22	86	10	00	20	23	43	FD	21	FE	23	188
27A0	C9	CD	D1	33	05	01	79	27	05	E9	D1	C9	2A	AE	10	7E
27B0	22	B2	10	C9	CD	70	24	20	05	1F	FF	18	0E	20	1C	196
27C0	25	28	3A	FE	23	28	40	FE	26	28	58	FE	21	28	6F	168
27D0	2C	20	03	13	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B
27E0	18	02	13	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E	1B	1E
27F0	C3	FE	BC	28	03	1B	1E	E2	EF	07	13	18	87	13	CD	01
Sum: 92 BA DC 86 88 6F 5E 64 15 2D 54 BF FF DD 66 AB 158																

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
2800	33	7D	32	4A	15	18	AD	13	1A	13	FE	32	28	0D	3A	137

2810	C2	A3	2A	D1	33	CD	D4	1C	1B	99	CD	D1	33	CD	D8	1A
2820	1C	1B	91	13	CD	D1	33	CD	D4	1C	1B	99	CD	D1	33	CD
2830	22	CA	B4	27	FE	00	20	02	18	C9	EF	03	1E	1F	1E	1B
2840	D1	33	EB	08	4A	1C	EB	C3	B4	27	CD	00	28	CD	6A	16A
2850	CD	D1	33	7D	FE	19	02	D4	2A	32	04	00	C9	CD	D1	33
2860	70	FE	50	02	D4	2A	32	03	00	C9	ED	1C	1E	22	20	1A
2870	11	13	1A	13	FE	22	28	09	FE	00	CA	A3	2A	EF	03	1B
2880	F1	13	1C	CD	60	33	E5	05	3A	03	00	F5	CD	4A	1C	F1
2890	SF	CD	ED	1C	1A	FE	24	20	05	CD	77	17	18	10	FE	20
28A0	20	09	13	CD	71	1C	CD	FF	15	1B	03	CD	71	1C	D1	4A
28B0	40	E1	71	23	70	CD	ED	1C	FE	2C	10	13	1B	CA	70	7D
28C0	24	21	13	13	20	04	36	0D	1C	1E	22	C2	D4	2A	13	1B
28D0	20	03	1B	13	20	03	18	1B	03	07	18	36	07	23	70	7D
28E0	FE	01	28	02	18	EA	3A	7E	21	11	13	CD	22	12	13	1B
28F0	1D	ED	A3	14	13	EA	3A	1D	B7	ED	42	23	22	12	13	1B
Sum	75	03	D1	91	6C	9E	AB	C2	47	54	C3	CE	3A	26	B9	42

ダンプ・リスト

Sum:	E6	52	EA	BF	40	2F	97	7D	0E	9E	D5	07	B4	41	E3	E5	179
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3700	CD	60	33	CD	77	36	C9	CD	67	36	E5	CD	74	36	C1	D5	1FF
3710	11	31	37	D5	C5	C9	CD	D1	33	EB	C3	12	1C	EB	C3	0E	145
3720	1C	CD	67	36	22	AC	15	CD	74	36	22	AE	15	D5	CD	D0	137
3730	17	D1	C9	D5	CD	3E	17	0D	20	2A	2A	2A	20	56	61	72	19C
3740	69	61	62	6C	65	73	20	2A	2A	2A	00	00	11	00	3C	2A	172
3750	B6	1D	B7	ED	52	20	04	D1	EF	07	C9	1A	13	FE	00	2B	1DD
3760	04	EF	03	1B	F6	EF	27	11	11	00	1A	13	6F	1A	13	67	16C

3770	3A	A4	15	F5	E5	21	A4	15	36	06	E1	E5	CD	0B	17	EF	1B7
3780	27	20	24	00	E1	EF	19	F1	32	A4	15	EF	07	EF	0C	FE	11F
3790	20	28	FA	EF	0D	20	B8	D1	C9	00	00	00	00	00	00	00	1B0
37A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
37B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
37C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
37D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
37E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
37F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100

Sum: B5 08 E9 02 AB 9B 82 5B 89 5C DA B8 2C 5E 31 CB 14B

WICS 303

3 Super COMPILER

キャリアラボ 山本耕司

今回はWICS 80Bのコンパイラを発表します。気の早い方はすでに80K/C用のコンパイラを80Bに移植して使っているかもしれませんが、今回発表するバージョンはさらに最適化をすすめているものです。

入力方法

オブジェクトのダンプ・リストをTS-1000モニタから打ち込んでください。

ソース・リストをインタプリタで打ち込んでもコンパイルできないので注意してください（インタプリタでは動きません）。

コンパイラの機能強化などを行なう場合はコンパイラによりソース・リストを再コンパイルすることになります。

強化された最適化機構について

- ① 256以下の定数を乗じる場合、すべてADD命令で展開されます。

例1. $A = A * 10$

```
LD HL,(VAR.A)
LD E,L
LD D,H
ADD HL,HL
ADD HL,HL
ADD HL,DE
ADD HL,HL
LD (VAR.A),HL
```

例2. $A = A * 256$

```
LD HL,(VAR.A)
LD H,L
LD L,0
LD (VAR.A),HL
```

ただし、定数は演算子“*”の右側に書いてください。なお、256以上の定数を乗じる場合はランタイム・サブルーチンをコールするように展開されます。

- ② 2^n で表わされる定数で割る場合、
n 個分の算術右シフト命令に置き替えます。ただし、n ≤ 8 の場合はさらに技巧を凝らしています。

例3. $A = A / 8$

```
LD HL,(VAR.A)
SRA H
RL L
SRA H
RL L
SRA H
RL L
LD (VAR.A),HL
```

例4. $A = A / 512$

```
LD HL,(VAR.A)
LD A,H
LD L,H
ADD A,A } ÷256の処理
SBC A,A
LD H,A
SRA H } ÷2の処理
RR L
LD (VAR.A),HL
```

÷256の処理は符号を保存する必要から、多少長くなっています。

- ③ 2^n で表わされる定数のMODをとる場合

例5. $A = \text{MOD}(A, 16)$

```
LD HL,(VAR.A)
LD DE,15
CALL @AND
LD (VAR.A),HL
```

$2^n - 1$ とのANDをとるだけで済ませています。

マシン語データの記述方法について

BASEと同じ表現方法“/”で行ないます。すなわち、

/ (マシン語データ)

＜マシン語データ＞

- 任意個の16進数（\$マークはつけない）
- Lで始まる行番号（DEFWと同じ機能）
- Vで始まる変数名（DEFWと同じ機能）
- “（クォーテーション）でつくられた任意個の文字列（DEFMと同じ機能）

例6. あるマシン語サブルーチンをコールしたのち、Aレジスタの内容をAccという変数に代入する。

```
10 CALL SUBR
20 / (6F 26 00 22 V.ACC)
```

例7. あるマシン語ルーチンをコールしたのち、キャリーフラグが立っていれば“ERR”の表示を出す。

```
10 CALL SUBR
20 / (D2 L.40)
30 PRINT "ERR"/
40 END
```

例8. テーブルによるデータ交換（ここでは0～15までの数字をASCIIコードに変換している）。

```
10 / (21 L.1000 22 V.TABLE)
20 ASCII=TABLE [DATA]
30 RETURN
1000 / ("0 1 2 3 4 5 6 7 8 A B C D E F")
```

以上のように、このコンパイラではソース・リスト中に自由にマシン語やデータをばらまくことができます。

また、ループの内側など、最も実行回数の多い行をマシン語化することにより、完全にアセンブラと同等の速度を

得ることもできます。ただし、インタープリタ・モードでは動かないので注意してください。

サンプル・プログラム

1 万桁の円周率を求めるプログラムを示します（リスト3）。1000桁ならば1分足らず、1万桁でも2時間で求まります。

終わりに

このコンパイラには、BASEと同じようなOFFSE T機能やROM化のための変数エリア設定（RAMAD）など紙面の都合により説明できなかった多くの機能を含んでいますが、これらの機能を使いこなすことのできる読者ならばソース・リストをごらんになっていただければわかってもらえると思い、ごちゃごちゃと説明するよりもソース・リストを載せました。このコンパイラの最適化でまだ不足と思った方はさらに最適化を試みてください。

リスト1 WICSコンパイラ・ソース・リスト

```
10 '*****
20 '----- WICS Compiler Ver1.80 -----
30 ' Copyright(C) by carrysoft
40 ' written by K.Yamamoto
50 ' k.T.Sasaki
60 ' 26-Nov-1981
70 '*****
80 LIB=1500:OFFSET=0:RAMAD=0
100 '(EF27 00)*** WICS Compiler Ver 1.80 ***0000 00)
110 INPUT "Source Program Address: ",SRC:GOSUB 10000
130 INPUT "Object Program Address: ",OBJ:GOSUB 10000
150 INPUT "Working area Address: ",WK:GOSUB 10000
170 PRINT "Address Table(Y/N):",DPSM=(GET="Y")
180 PRINT "Pass (2/4):",PSM=(GET="2")
200 CALL #1607
210 PASS=0:ERN=PASS
220 GOSUB 260
230 IF PSM=0 THEN GOSUB 260
240 GOTO 10160
250 '
260 GOSUB 270
270 PASS=PASS+1:PRINT "3, /, ",PASS, /
280 GOTO 1000
490 '
500 '--- Space skip ---
510 A=S(0):IF A<0) * RETURN
520 S=S+1 GOTO 510
530 '
540 '--- Call Runtime ---
550 CAL=CAL+LIB
560 '--- Call Generate ---
570 GOSUB 900:(01 CD):(B+):CAL:RETURN
580 '
590 '--- Check Operator ---
600 GOSUB 500:IF A<0)80 THEN A=0:RETURN
610 A=S(1):RETURN
620 '
630 '--- Jump Address Calc ---
640 IF MOD(PASS,2)<0 THEN JA=0:RETURN
645 T2=TABLE-4
649 '18 if T1-2)<HL then T2=T-B
650 '(24V,T 11FCFF19 7E23666F E82AV,HL B7E052 380A 2AV,T 11FBFF19 22V,T2)
660 T2=T2+4:IF T2(0)=HL THEN JA=T2(1):RETURN
670 IF T2(1)<0)FF GOTO 660
680 ERN=1:GOTO 14000
690 POP B1
```

```
700 '--- formula Optimize ---
710 OPT=0
720 IF B-B1<0)3 GOTO 760
730 A=B1(0)
740 IF A=421 GOTO 770
750 IF A=42A GOTO 790
760 GOSUB 900:(02 EBE1):RETURN
770 B1(1)=B1(0)-B(1):B=B-1
780 OPT=1:RETURN
790 B1(1)=B(0):B(0)=B(0):RETURN
800 ' LHL 0omit
810 IF (B-OLDLET=3) AND (OLDVAR=VA):RETURN
820 '--- LHL VA
830 GOSUB 900:(01 2A):GOTO 860
840 '--- SHL VA
850 GOSUB 900:(01 22)
860 [B+]=VA:RETURN
900 '--- LONG CODE GENERATOR ---
910 '(E3 4E 23 0600 ED5B V.8 ED80 ED53 V.8 E3 C9)
990 '
1000 '----- Pass 1, 2 -----
1010 S=SRC:B=OBJ:TABLE=WKG+120:T=TABLE:FSX=T
1020 VTBL=22000 VTBLEND=VTBL
1030 LMD=SI(0)256+S(1)
1040 IF SI(0)=FF GOTO 1090
1050 ATF=0:GOSUB 10100
1060 T(0)=LMD:T(1)=B:T=T+4
1070 GOSUB 1150:ATF=1:GOSUB 10100
1080 GOTO 1030
1090 '
1100 IF FSX<TABLE THEN PRINT "NEXT not found":GOTO 14040
1110 T(0)=65535:T(1)=B:T=T+4
1120 GOSUB 900:(01 C3):(B+):LIB+6
1130 IF MOD(PASS,2)<0) THEN BEND=8
1140 RETURN
1150 '----- 1 line Compile -----
1160 S=S+3:OLDLET=0:IXI=0
1170 IF SI(0)=0 THEN S=S+1:RETURN
1180 GOSUB 1200 :1 STEP
1190 GOTO 1170
1200 '----- 1 step Compile -----
1210 GOSUB 500:"Space Skip
1220 IF A=" THEN S=S+1:GOSUB 500
1230 IF A=0) GOTO 2000:"Statement
1240 '----- LET ---
1250 GOSUB 500:IF A="L" GOTO 1500
```

```

1255 IF A="*" GOTO 9000
1260 GOSUB 8000
1270 GOSUB 500
1280 IF A="I" GOTO 1340
1290 IF A="(" GOTO 1410
1300 GOSUB 590:IF A<>484 GOTO 14000
1310 PUSH VA:GOSUB 4980:POP VA
1320 ' - SHLD Variable -
1330 OLDLET=B:OLDVAR=VA : GOTO 840
1340 ' - Store Indexed I -
1350 S=S+1:GOSUB 800
1360 GOSUB 1490:GOSUB 6660
1370 GOSUB 1450:GOSUB 900'(01 73):IF OPT<>1 RETURN
1380 B=B-2:B(-2)=B36:RETURN
1390 '
1400 ' - Store Indexed 2 -
1410 S=S+1:GOSUB 800
1420 GOSUB 6480
1430 GOSUB 1440:GOSUB 900'(03 732372):RETURN
1440 GOSUB 7410:GOTO 1460 'J'
1450 GOSUB 500:GOSUB 6600 'J'
1460 GOSUB 590:IF A<>484 GOTO 14000 'S='
1470 S=S+2
1480 'H DE=Eval H
1490 GOSUB 900'(01 E5):PUSH B:GOSUB 5000:GOTO 690
1500 ' - AUTO INDET -
1510 S=S+1:GOSUB 8000:GOSUB 590
1520 IF A<>489 GOTO 14000
1530 S=S+2
1540 GOSUB 500:IF A="*" GOTO 1600
1550 GOSUB 590:IF A<>489 GOTO 14000
1560 S=S+2
1570 GOSUB 1650
1580 GOSUB 900'(02 7223) :GOTO 1320
1590 '
1600 GOSUB 1650
1610 IF B(-93)<421:GOTO 1320
1620 B=B-9:ATDAT=B(1):GOSUB 800:IF B(-3)<422 THEN B=B-3
1630 GOSUB 900'(01 361:[B+3]:ATDAT:GOSUB 900'(01 23):GOTO 1320
1640 '
1650 GOSUB 6600 'J'
1660 GOSUB 590:IF A<>484 GOTO 14000
1670 PUSH VA:GOSUB 4980:GOSUB 900'(01 E8):POP VA
1680 GOSUB 820:GOSUB 900'(02 7323) :RETURN
1690 '
2000 '##### Statement #####
2010 S=S+1 A=(S+1)
2020 IF A<20 GOTO 14000
2030 IF A<30 RETURN
2040 ON A-42F GOTO 2130,2120,2380,2490,2600,2620,2660,2840
2050 ON A-437 GOTO 2860,3000,3020,3050,3240,2110,3320,3320
2060 ON A-43F GOTO 3330,3340,3360,3400,3440,3490,3540,3580
2070 ON A-447 GOTO 8420,8450,3920,2110,2110,2110,2110,2110
2080 ON A-44F GOTO 2110,3610,3640,3670,3700,3730,3760,3790
2090 ON A-457 GOTO 3820,3840,3860,3900,3920,3950,3970,4060
2100 ON A-45F GOTO 4090
2110 GOTO 14000
2120 RETURN
2130 ' -IF-
2140 GOSUB 5000:OP1=0:JIF=4C2
2150 GOSUB 2330:IF OP1=0 GOTO 2260
2160 IF OP1=1 THEN B=B-3:GOSUB 900'(03 87E052) :GOTO 2280
2170 B=B-6:A=B(1)
2180 IF B-1X1<3 GOTO 2230
2190 B=B-3:IF 1X2<0 THEN GOSUB 900'(01 7E)
2200 IF A=0 THEN GOSUB 900'(01 87):GOTO 2280
2210 GOSUB 900'(01 FE) :[B+]=A:GOTO 2280
2220 '
2230 GOSUB 900'(01 7D)
2240 IF A=0 GOTO 2270
2250 GOSUB 900'(01 D6):[B+]=A:GOTO 2270
2260 GOSUB 900'(01 7D)
2270 GOSUB 900'(01 84)
2280 GOSUB 590:IF A<37 THEN S=S+2:OP=JIF:GOTO 2870 ' -GOTO'

```

```

2290 IF A<39 THEN S=S+2:[B+]=JIF-2:RETURN ' -RETURN'
2300 [B+]=4EA-JIF:A=T(1)-B-1:[B+]=A ' -THEN'
2310 IF (PASS=2) AND (A)>7F THEN ERN=3 GOTO 14000
2320 '
2330 IF B(-3)<4CD RETURN
2340 A=B(-1):IF A<4C(LIB+4D5) AND A<4C(LIB+4DE) RETURN
2350 OP1=1:IF A=LIB+4D5 THEN JIF=4CA
2360 IF (B(-6)<411)+(B(-4)<3C0) RETURN
2370 OP1=2:RETURN
2380 ' -FOR-
2390 GOSUB 8000:PUSH VA
2400 GOSUB 590:IF A<>484 GOTO 14000:' -FOR
2410 GOSUB 4980:POP VA:GOSUB 840:FSTK=FSTK-10:FSTK(4)=VA
2420 GOSUB 590:IF A<>44E GOTO 14000:' -TO
2430 PUSH B:GOSUB 4980:POP B1
2440 IF (B-B1)<3 AND (B1(0)<421) THEN FSTK(3)=B(-1):B=B1:FSTK(2)=0:GOTO 2460
2450 GOSUB 8150:GOSUB 840:FSTK(2)=VA
2460 GOSUB 590:IF A<>44F THEN FSTK(1)=1:GOTO 2480:' -STEP
2470 GOSUB 4980:GOSUB 8150:GOSUB 840:FSTK(1)=VA
2480 FSTK(0)=B:RETURN
2490 ' -NEXT-
2500 GOSUB 500:IF (A)=*A AND (A)=*2)GOSUB 8000
2510 VA=FSTK(4):GOSUB 820
2520 IF FSTK(1)=1 THEN [B+]=423 GOTO 2540
2530 GOSUB 900'(02 E05B) :[B+]=FSTK(1):GOSUB 900'(01 19)
2540 GOSUB 840
2550 GOSUB 900'(01 E8):VA=FSTK(2):IF VA=0 THEN GOSUB 900'(01 21):VA=FSTK(3):
GOSUB 860:GOTO 2570
2560 GOSUB 820
2570 GOSUB 900 '104 B7ED52F2):VA=FSTK(0)+OFFSET:GOSUB 860
2580 FSTK=FSTK+10:IF FSTK>TABLE THEN ERN=2 GOTO 14000
2590 RETURN
2600 ' -REPEAT-
2610 GOSUB 900'(01 CD):VA=B+OFFSET+2:GOTO 860
2620 ' -UNTIL-
2630 GOSUB 5000
2640 GOSUB 900'(07 7C85 E1 2002 E5E9)
2650 RETURN
2660 ' -ON-
2670 GOSUB 5000:GOSUB 590:S=S+2
2680 IF A<37 GOTO 2750 ' -ONGOSUB
2690 IF A<43B GOTO 14000 ' -ONGOSUB
2700 GOSUB 2780
2710 GOSUB 900'(04 110000 D5)
2720 GOSUB 2800
2730 GOSUB 900'(01 C9):POP B2:B2(1)=B+OFFSET:GOTO 2760
2740 '
2750 GOSUB 2780 :GOSUB 2800 :POP B2
2760 B2(0)=B-B2-1:RETURN
2770 '
2780 GOSUB 900'(03 7C B7 20):PUSH B:B=B+1
2790 RETURN
2800 GOSUB 900'(01 2D):OP=4CA:GOSUB 2870:GOSUB 500
2810 IF A="," THEN S=S+1:GOTO 2800
2820 RETURN
2830 ' -GOTO
2840 OP=4C3:GOTO 2870
2850 ' -GOSUB
2860 OP=4CD
2870 [B+]=OP:GOSUB 500:SR=S
2880 A=S(0):RELFLG=(A)>480:S(0)=A AND 47F
2890 GOSUB 6170:'DECIMAL
2900 GOSUB 630:[B+]=JA+OFFSET
2910 IF PASS=1 RETURN
2920 IF PASS=2 GOTO 2980
2930 IF RELFLG=0 RETURN
2940 IF PASS=3 THEN SR(0)=SR(0)+480
2950 B=B-3:A=18:B(0)<4C3 THEN A=B(0)-4A2
2960 [B+]=A:[B+]=JA-B-1:RETURN
2970 '
2980 IF (OP=4CD)+PSW+(ABS(JA-B)>47D) RETURN
2990 SR(0)=SR(0)+480 RETURN
3000 ' - RETURN -
3010 GOSUB 900'(01 C9):RETURN

```

リスト1 WICSコンパイラ・ソース・リスト

```

3020 *-REM-
3030 IF S(0)=400 RETURN
3040 S=S+1:GOTO 3030
3050 *-PRINT-
3060 GOSUB 500:IF (A=":)+(A=0) RETURN
3070 IF A=22 THEN CAL=1E:GOSUB 540:GOSUB 3160:GOTO 3060
3080 IF A="2" THEN GOSUB 4990:(B+)=22:(B+)=L1B+AA:GOTO 3060
3090 IF A="1" GOTO 3190
3100 IF A="1" THEN GOSUB 4990:CAL=4C:GOTO 3230
3110 IF A="1" THEN GOSUB 4990:CAL=1B:GOTO 3230
3120 IF A="," THEN S=S+1:GOTO 3060
3130 GOSUB 590 IF A=18C THEN S=S+2:CAL=9:GOTO 3230
3140 IF A=189 THEN GOSUB 4990:CAL=12:GOTO 3230
3150 GOSUB 5000:CAL=1F:GOTO 3230
3160 S=S+1:A=S(0):IF (A<22) AND (A<40) THEN (B+)=A:GOTO 3160
3170 GOSUB 900'(0100):IF A=22 THEN S=S+1
3180 RETURN
3190 S=S+1:A=(S+):IF A="2" THEN GOSUB 5000:CAL=1B:GOTO 3230
3200 IF A="4" THEN GOSUB 5000:CAL=15:GOTO 3230
3210 GOTO 14000
3220 *-
3230 GOSUB 540:GOTO 3060
3240 *-INPUT-
3250 GOSUB 500:IF A<22 GOTO 3290
3260 CAL=1E:GOSUB 540
3270 S=S+1:A=S(0):IF A<22 THEN (B+)=A:GOTO 3270
3280 GOSUB 900'(0100):S=S+2
3290 GOSUB 8000:CAL=21:GOSUB 540:GOSUB 1320
3300 GOSUB 500:IF A="," THEN S=S+1:GOTO 3290
3310 RETURN
3320 *-STOP-
3330 CAL=6 GOTO 540 *-MDM
3340 CAL=0 GOTO 560 *-BYE
3350 *-CALL
3360 PUSH B:GOSUB 5000
3370 POP B:IF (B-B1)=31 AND (B1(0)=21) THEN B1(0)=4C:D:RETURN
3380 (B+)=01:(B+)=B+4:GOSUB 900'(02 C5E9)
3390 RETURN
3400 *-PUSH-
3410 GOSUB 5000:CAL=0114:GOSUB 540
3420 GOSUB 500:IF A<22," RETURN
3430 S=S+1:GOTO 3410
3440 *-POP-
3450 GOSUB 8000:CAL=012B:GOSUB 540
3460 GOSUB 1320
3470 GOSUB 500:IF A<22," RETURN
3480 S=S+1:GOTO 3450
3490 *-OUT
3500 GOSUB 3520:GOSUB 900'(03 4D E059)
3510 RETURN
3520 *-2 PARAMETERS HL,DE-
3530 GOSUB 5000:GOSUB 7440:GOTO 1490
3540 *-CURSOR
3550 GOSUB 3520
3560 GOSUB 900'(04 63 22 0300):RETURN
3580 *-POKE
3590 GOSUB 3520:GOSUB 900'(01 73)
3600 RETURN
3610 *-SPOKE
3620 GOSUB 4170:GOSUB 4210
3630 CAL=13C:GOTO 540
3640 *-CURH
3650 GOSUB 5000
3660 VA=40003:GOTO 3690
3670 *-CURV
3680 GOSUB 5000:VA=40004
3690 GOSUB 900'(02 7D 32):GOTO 860
3700 *-PLOT
3710 GOSUB 4170:GOSUB 4210
3720 CAL=12A:GOTO 540
3730 *-LINE
3740 GOSUB 4290:GOSUB 4210
3750 CAL=133:GOTO 540
3760 *-PUT2

```

```

3770 GOSUB 4250:GOSUB 4210
3780 CAL=139:GOTO 540
3790 *-GET2
3800 GOSUB 4250
3810 CAL=136:GOTO 540
3820 *-PRGDE
3830 GOSUB 5000:VA=40036:GOTO 3690
3840 *-BELL
3850 GOSUB 900'(02 EF17):RETURN
3860 *-MUSIC
3870 GOSUB 4110
3880 BE-(4)=40
3890 GOSUB 900'(02 EF14):RETURN
3900 *-TENPO
3910 GOSUB 5000:GOSUB 900'(03 7D EF1B):RETURN
3920 *-SOUND
3930 GOSUB 3520:GOSUB 900'(04 4B42 EF13):RETURN
3950 *-TIME=
3960 GOSUB 5000:GOSUB 900'(03 EB EF15):RETURN
3970 *-GRAPH
3980 GOSUB 500
3990 IF A="1" THEN GOSUB 4990:CAL=157:GOTO 4050
4000 IF A="0" THEN GOSUB 4990:CAL=15A:GOTO 4050
4010 IF A="C" THEN S=S+1:CAL=15A:GOTO 4050
4020 IF A="F" THEN S=S+1:CAL=15D:GOTO 4050
4030 IF A="," THEN S=S+1:GOTO 3970
4040 RETURN
4050 GOSUB 540:GOTO 3970
4060 *-CIRCLE
4070 GOSUB 4310:GOSUB 4270
4080 GOSUB 4210:CAL=42D:GOTO 540
4090 *-PAINT
4100 GOSUB 4170:CAL=46D:GOTO 540
4110 *-JRE "....."
4120 GOSUB 500:IF A<22 GOTO 14000
4130 (B+)=11B:PUSH B:B=B+1:GOSUB 3160
4140 POP B:B1(0)=B-B1-1:(B+)=11:(B+)=B1+OFFSET+1:RETURN
4150 *-1 PARAMETER A",
4160 GOSUB 5000:GOTO 7440
4170 *-2 PARAMETER PX,PY
4180 GOSUB 4150:VA=L1B+4AC:GOSUB 840:'PX
4190 GOSUB 5000:VA=L1B+4AE:GOSUB 840:'PY
4200 RETURN
4210 *-N PARAMETER
4220 GOSUB 500 IF A<22," THEN GOSUB 900'(03 21 01 00):GOTO 4240
4230 S=S+1:GOSUB 5000
4240 VA=L1B+4AB:GOTO 840:'PM
4250 *-5 PARAMETERS (X,Y,I1,Y1,AD)
4260 GOSUB 4290
4270 GOSUB 7440:GOSUB 5000:VA=L1B+4B4:GOSUB 840:'PAD
4280 RETURN
4290 *-4 PARAMETERS (X,Y,X1,Y1)
4300 GOSUB 4180:GOSUB 7440:'PX,PY
4310 GOSUB 4150:VA=L1B+4B0:GOSUB 840:'PX1
4320 GOSUB 5000:VA=L1B+4B2:GOSUB 840:'PY1
4330 RETURN
4900 *-
4970 S=S+2:GOSUB 900'(01 E5):PUSH A,B:RETURN
4980 S=S+1
4990 S=S+1
5000 *-If formula evaluation **
5010 *-HL:=Eval
5020 GOSUB 5180
5030 GOSUB 590:'800?
5040 IF (A<80)+(A)=88B) RETURN
5050 GOSUB 4970:GOSUB 5180
5070 GOSUB 690:'Optimize
5080 POP A:ON A-97F GOTO 5100,5100,5110,5110,5120,5130,5140,5150
5090 GOTO 14000
5100 '(3E E7 01) 'E="
5110 '(3E F2 01) 'F="
5120 '(3E B5 01) 'B="
5130 '(3E DE 01) 'D="
5140 '(3E EB 01) 'E="

```

リスト 1 WICS コンパイラ・ソース・リスト

```

5150 '(3E F1 ) )'
5160 '(6F 26 00 22 V.CAL) :GOSUB 540:GOTO 5030
5170 '
5180 '- EVAL PLUS -
5190 GOSUB 5320
5200 GOSUB 590:'80+?
5210 IF (A(88B)+(A)80A) RETURN
5230 GOSUB 4970:GOSUB 5320
5240 GOSUB 690
5250 POP A:ON A-807 GOTO 5270,5280,5290
5260 GOTO 14000
5270 CAL=900C7:GOSUB 540:GOTO 5200 :'DR
5280 GOSUB 6660:GOTO 5200 :'+'
5290 IF OPT=1 THEN B(-)=B(-):GOSUB 6660:GOTO 5200: '-'
5300 GOSUB 900'(03 B7 ED 52):GOTO 5200
5310 '
5320 '- EVAL MUL -
5330 GOSUB 5690
5340 GOSUB 590:'80+?
5350 IF (A(88B)+(A)=80F) RETURN
5370 GOSUB 4970:GOSUB 5690
5380 GOSUB 690
5390 POP A:ON A-88A GOTO 5410,5530,5590,5600
5400 GOTO 14000
5410 IF OPT<1 GOTO 5510
5420 MCONST=B(-1)
5430 IF MCONST<256 GOTO 5510
5440 IF MCONST<1 GOTO 5510
5460 B=B-3
5480 IF MCONST=2 THEN (B+)=829 GOTO 5340
5490 IF MCONST=256 THEN GOSUB 900'(03 65 2E 00 ) :GOTO 5340
5500 GOSUB 5620:GOTO 5340
5510 CAL=801B7:GOTO 5610 :'8'
5520 '- Divide
5530 IF OPT=0 GOTO 5580
5540 MCONST=B(-1):GOSUB 5740:IF MF=0 GOTO 5580
5545 GOSUB 5550:GOTO 5340
5550 B=B-3:IF MF<0 THEN GOSUB 900'(05 7C AC B7 9F 67):MF=MF-B
5560 MF=MF-1:IF MF=0 RETURN
5570 GOSUB 900 '(04 CB2C CB1D) :GOTO 5560
5580 CAL=80131:GOTO 5610 : '/'
5590 CAL=900C0:GOTO 5610 : 'AND
5600 CAL=900CE : 'IOR
5610 GOSUB 540:GOTO 5340
5620 '### Multiply Optimize
5630 MCNT=9 : GOSUB 5740:IF MF=0 THEN GOSUB 900'(02 5D 54)
5640 GOSUB 5810:MCNT=MCNT-1:IF BIT=0 GOTO 5640
5650 GOSUB 5810:MCNT=MCNT-1:IF MCNT=0 RETURN
5660 GOSUB 900'(01 29):IF BIT<0 THEN (B+)=819
5670 GOTO 5650
5680 '
5690 '- EVAL PAR -
5700 GOSUB 500
5710 IF A<'(' GOTO 6000
5720 GOSUB 4990
5730 GOTO 7410:)'
5740 '-MCONST TEST-
5750 MB=1:MF=0
5760 FOR MI=0 TO 15
5770 IF MCONST=MB THEN MF=MI+1
5780 MB=MB*2
5790 NEXT
5800 RETURN
5810 ' MSB Check
5820 MCONST=MCONST*2:BIT=MCONST AND 8100:MCONST=MCONST AND 8FF
5830 RETURN
5890 '
6000 '## Term evaluation ##
6010 ' HL:=Eval
6020 GOSUB 500
6030 IF (A)=0'1 AND (A<'9') GOTO 6150 :'Decimal Const
6040 IF A=800 GOTO 7000 :'Function
6050 IF A='(' GOTO 6320 :'Auto Index
6060 IF A='*' GOTO 6230 :'Hex
6070 IF A=422 GOTO 6310 :'Chr
6080 GOSUB 8000 :'Var
6090 GOSUB 800
6100 GOSUB 500
6110 IF A='(' GOTO 6370 :'Index 1
6120 IF A='(' GOTO 6430 :'Index 2
6130 RETURN
6140 '
6150 '- Decimal Constant -
6160 GOSUB 6170:GOTO 6330
6170 GOSUB 500
6180 IF (A<'0')+(A>'9') GOTO 14000
6190 HL=A-'0'
6200 S=S+1:A=S(0):IF (A<'0')+(A>'9') RETURN
6210 HL=HL*10+A-'0':GOTO 6200
6220 '
6230 '- Hexadecimal -
6240 S=S+1:GOSUB 6250:GOTO 6330
6250 GOSUB 500
6260 GOSUB 6340:IF A=16 GOTO 14000
6270 HL=A
6280 S=S+1:A=S(0)
6290 GOSUB 6340:IF A=16 RETURN
6300 HL=HL*16+A:GOTO 6280
6310 '- Character -
6320 S=S+1:HL=[S]:IF [S]<422 GOTO 14000
6330 GOSUB 900'(01 21):VA=HL:GOTO 860
6340 A=A-'0':IF (A)=0) AND (A<10)RETURN
6350 A=A-7 :IF (A)=10) AND (A<16)RETURN
6360 A=16:RETURN
6370 '- Indexed Variable 1 -
6380 S=S+1
6390 GOSUB 1490:GOSUB 6660
6400 I1=8:I12=1:GOSUB 7270 :'H=0 L=(HL)
6410 GOTO 6600 :)'
6420 '
6430 '- Indexed Variable 2 -
6440 S=S+1
6450 GOSUB 6480
6460 GOSUB 900 '(04 7E23666F)
6470 GOTO 7410 :)'
6480 GOSUB 1490:IF OPT<1 GOTO 6500
6490 B(-)=B(-)*2:GOTO 6660
6500 GOSUB 900'(02 1919):RETURN
6510 '
6520 '- Auto Indexed -
6530 S=S+1:GOSUB 8000:GOSUB 590
6540 IF A<'89 GOTO 14000
6550 S=S+2
6560 GOSUB 500:IF A=')' GOTO 6620
6570 GOSUB 590:IF A<'89 GOTO 14000
6580 S=S+2
6590 GOSUB 800:GOSUB 900'(04 5E23 5623):GOSUB 840:GOSUB 900'(01 EB)
6600 IF [S]<'(' GOTO 14000
6610 RETURN
6620 GOSUB 800:GOSUB 900'(02 7E23):GOSUB 840
6630 I1=B:I12=0:GOSUB 900'(01 6F):GOSUB 7280
6640 GOTO 6600:)'
6650 '
6660 ' Optimize 'Add(Subtract) Constant
6670 IF OPT<1 GOTO 6710
6680 BM=B(-1)
6690 IF (BM)=0) AND (BM<4) THEN OP=823:GOTO 6720
6700 IF (BM<0) AND (BM>-4) THEN OP=828:GOTO 6720
6710 GOSUB 900'(01 19):BM=1:RETURN
6720 BM=ABS(BM):B=B-3
6730 IF BM=0 RETURN
6740 (B+)=OP
6750 BM=BM-1:GOTO 6730
6790 '
7000 '### Function ###
7010 GOSUB 590:S=S+2
7020 IF A<'8A GOTO 7070
7030 PUSH B:GOSUB 5690:POP B

```

```

7040 IF (B-B1=3) AND (B1(0)=421) THEN B(-1)=-B1(-1):RETURN
7050 CAL=400FF:GOTO 540 :'-
7060 '
7070 ON A-467 GOTO 7510,7540,7560,7100,7100,7100,7590,7410
7080 ON A-46F GOTO 7120,7130,7140,7190,7200,7210,7220,7380
7090 ON A-477 GOTO 7250,7300,7320,7340
7100 GOTO 14000
7110 '
7120 CAL=424:GOTO 540 : 'GET
7130 CAL=427:GOTO 540 : 'INKEY
7140 GOSUB 7470:IF OPT=0 GOTO 7180
7150 NCONST=B(-1)
7160 GOSUB 5740:IF MF=0 GOTO 7180
7170 B(-1)=NCONST-1:CAL=40C0:GOTO 540:'.AND.
7180 CAL=41B2:GOTO 540 : 'MOD(
7190 GOSUB 7500:CAL=40FB:GOTO 540 : 'ABS(
7200 GOSUB 7500:CAL=4107:GOTO 540 : 'SGN(
7210 GOSUB 7500:CAL=41BB:GOTO 540 : 'AND(
7220 '-INP(
7230 GOSUB 7500:GOSUB 900:(05 4D ED6B 2600):RETURN
7240 '
7250 '-PEEK(
7260 GOSUB 7500
7270 GOSUB 900:(01 4E) : 'MOV L,M
7280 GOSUB 900:(02 26 00):RETURN: 'MVI H,0
7290 '
7300 PUSH 423:GOTO 7330: 'INC(
7310 '
7320 PUSH 42B : 'DEC(
7330 GOSUB 8000:GOSUB 7410:GOSUB 820:POP A:(B+)=A:GOTO 840
7340 '-USR(
7350 GOSUB 7470:(B+)=401:(B+)=B+6
7360 GOSUB 900 :!04 C5E5 EBE9)
7370 RETURN
7380 '-ADRS(
7390 GOSUB 8000:(B+)=421:GOSUB 860
7400 '
7410 GOSUB 500:IF (S+)=) RETURN
7420 GOTO 14000
7430 '
7440 GOSUB 500:IF (S+)=, RETURN
7450 GOTO 14000
7460 GOSUB 5000:GOTO 7410
7470 '- 2 PARAMETERS -
7480 GOSUB 5000:GOSUB 7440:GOSUB 1490:GOTO 7410
7490 '- 1 PARAMETER+)'
7500 GOSUB 5000:GOTO 7410
7510 '-SPEEK(
7520 GOSUB 7530:CAL=43F:GOTO 540
7530 GOSUB 4170:GOTO 7410
7540 '-TIME
7550 CAL=442:GOTO 540
7560 '-DOT(X,Y)
7570 GOSUB 7530
7580 CAL=430:GOTO 540: 'DOT
7590 '-CP4(
7600 GOSUB 4170:GOSUB 4210:GOSUB 7410:CAL=44B:GOTO 540
7610 '-CPI(
7620 GOSUB 4150:GOSUB 4110
7630 GOSUB 7410:CAL=44B:GOTO 540
7690 '
8000 '!! Get Variables address !!
8010 ' VR<-Address
8020 GOSUB 500
8030 '
8040 SI=S:VI=VTBL:VMD=0
8050 IF VI=VTBLEND GOTO 8100: 'MAKEVAR
8060 GOSUB 8220
8070 IF MATCH GOTO 8140
8080 S=SI:VMD=VMD+1:GOTO 8050
8090 '
8100 '!! Making Variables !!
8110 GOSUB 8310
8120 (VI+)= (S+):GOSUB 8350:IF VF GOTO 8120

```

```

8130 (VI+)=40D:VTBLEND=VI
8140 IF RAMAD(>) THEN VA=VMD+2:RAMAD:RETURN
8145 VA=VMD+2:BEND+OFFSET:RETURN
8150 '-WORK ADRS GEN.-
8160 IF PASS=1 THEN VA=0:RETURN
8170 VI=VTBL:VMD=1
8180 IF (VI+)<40D GOTO 8180
8190 IF VI<VTBLEND THEN VMD=VMD+1:GOTO 8180
8200 (VI+)= 'X':GOTO 8130
8210 '
8220 '!! Variables name matching !!
8230 MATCH=0
8240 GOSUB 8310
8250 IF S(0)<V(0) GOTO 8290
8260 S=S+1:VI=VI+1
8270 GOSUB 8350 :IF VF GOTO 8250
8280 IF V(0)=40D THEN MATCH=1:RETURN
8290 IF (VI+)=40D RETURN
8300 GOTO 8290
8310 '-Variable HEAD
8320 A=S(0):IF (A*"A")+(A*)"z") GOTO 14000
8330 IF (A*)"Z") AND (A*"a") GOTO 14000
8340 RETURN
8350 '-Variable SUFFIX
8360 VF=1 : A=S(0)
8370 IF (A)="A") AND (A!="Z") RETURN
8380 IF (A)="0") AND (A!="9") RETURN
8390 IF (A)="a") AND (A!="z") RETURN
8400 VF=0:RETURN
8410 '
8420 '-STA(
8430 GOSUB 4150:GOSUB 4110:GOSUB 7410
8440 CAL=44E:GOTO 540
8450 '-LINPUT
8460 GOSUB 5000:(B+)=EB:VA=400EF:GOTO 860
9000 '!!! MACHINE CODE !!!
9020 S=S+1:IF (S+)<)'(' GOTO 14000
9040 GOSUB 500:S=S+1
9050 IF A=)' RETURN
9060 IF A="L" GOTO 9200
9070 IF A="V" GOTO 9400
9080 IF A=422 GOTO 9600
9100 '
9120 GOSUB 6340:IF A=16 GOTO 14000
9130 AI=A:A=(S+)
9140 GOSUB 6340:IF A=16 GOTO 14000
9180 (B+)=A16+A:GOTO 9040
9200 '- L (line number)
9210 A=(S+):IF A(<)." GOTO 14000
9220 GOSUB 6170: 'DECIMAL
9230 GOSUB 630 : (B+)=JA+OFFSET:GOTO 9040
9400 '- V (variable name)
9410 A=(S+):IF A(<)." GOTO 14000
9420 GOSUB 8000:GOSUB 860:GOTO 9040
9600 '- "ASCII"
9610 A=(S+):IF A=40 GOTO 14000
9620-IF A=422 GOTO 9040
9640 (B+)=A:GOTO 9610
9990 '
10000 'rewrite (for printer use)
10010 : (22 V,PR4):CURSOR 23,PEEK(9004)-1
10020 PRINT "8",4 PR4,:RETURN
10030 '
10100 ' Object address table
10110 IF BPSW=0 RETURN
10115 IF MOD(PASS,2)<0 RETURN
10120 IF (PASS<4) AND (PSW=0) RETURN
10125 IF ATF=0 THEN PRINT 16, LMD, "8",4 PR4:RETURN
10130 PRINT "-8",4 PR4:GOTO 10350
10140 '
10150 '!!! Compile OK !!! !!!
10160 PRINT /, " XXX Complete ! XXX"/
10170 PRINT "Object size :",XS,BEND-OBJ
10180 PRINT " (8",4 OBJ, " - 8",4 BEND-1,")"/

```



```

10190 '
10200 '### Variables listing ###
10210 PRINT /," ### Variables ###"
10220 V1=VTL:VMD=0
10230 IF V1=VTL:END:GOTO 10330
10240 IF RAMAD=0 THEN PRINT #4 VMD42+BEND:GOTO 10250
10245 PRINT #4 VMD42+RAMAD
10250 BEND(VMD)=0:VMD=V1:"Variable Clear
10260 PRINT " :CC=8 REPEAT CC=CC-1
10270 PRINT # [V1]
10280 UNTIL V1(0)=800
10290 FOR C1=0 TO CC:PRINT " :NEXT
10300 V1=V1+1
10310 GOSUB 10350
10320 GOTO 10230
10330 PRINT //
10340 END
10350 - TAB 20 -
10360 PRINT # 111

```

```

10370 RETURN
13990 '
14000 '#### E R R O R #####
14010 BELL
14020 PRINT $5,/, " Compile error in LINE",+ LMD, /
14030 ON ERN+1 GOSUB 14050,14060,14070,14080
14040 PRINT /," ) Cancelled this compilation,":END
14050 PRINT " Syntax error,":RETURN
14060 PRINT " Under, LINE No.":RETURN
14070 PRINT " NEXT without FOR!":RETURN
14080 PRINT " Too long... THEN,":RETURN

```

リスト2 WICSコンパイラ・オブジェクト

```

adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D000 21 00 15 22 EE EE 21 00 00 22 ED EE 21 00 00 22 192
D010 EF EE EF 27 D0 2A 2A 20 57 49 43 53 20 43 6F 1A6
D020 6D 70 69 65 65 22 00 56 45 72 20 31 2E 38 30 20 18D
D030 2A 2A 2A 00 00 00 00 1E 15 53 6F 75 72 63 65 20 129
D040 50 72 6F 67 72 61 6D 20 41 64 64 72 65 73 73 3A 1F8
D050 00 21 15 22 F1 EE CD 57 EC CD 1E 15 4F 62 6A 12F
D060 65 63 74 20 50 72 6F 67 72 61 6D 20 41 64 64 72 1CF
D070 65 73 73 3A 00 00 22 F3 EE CD 53 EC CD 1E 186
D080 15 57 6F 72 68 69 6E 67 20 61 72 65 61 20 20 20 10F
D090 41 64 64 72 65 73 73 3A 00 00 21 15 22 F5 EE CD 1D5
D0A0 57 EC CD 1E 15 41 64 64 72 65 73 73 20 5A 61 62 140
D0B0 6C 45 59 2F 4E 29 00 CD 2A 15 11 59 00 CD D5 10A
D0C0 15 22 F7 EE CD 09 15 CD 1E 15 50 61 73 73 20 28 1E6
D0D0 32 2F 3A 29 00 CD 2A 15 11 32 00 CD D5 15 22 F9 1D9
D0E0 EE CD 7F 16 21 00 00 22 F5 EE 22 FD EE CD FD 00 17B
D0F0 2A F9 EE 7D B4 20 00 CD FD 00 C3 DB EC CD 00 D1 124

```

```

sums 39 CD C6 9D 04 7C CD DD 4C 9E A1 55 44 58 59 EB 146
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D100 2A FB EE 23 22 FB EE 21 03 00 22 8A 15 CD 09 15 128
D110 D1 15 3E 50 41 53 53 20 2A 2A FB EE CD 0F 1B1
D120 15 CD 09 15 C3 AC D2 2A 01 EF 4E 2A 00 22 FF EE 1FE
D130 7D D6 20 BA CD 2A 01 EF 23 22 01 EF 10 89 2A 03 164
D140 EF ED EB EE EE 19 22 03 CD 9B D2 01 CD 2A 03 172
D150 EF EB 2A 05 EF 73 73 72 22 22 05 EF C9 CD 27 D1 1C7
D160 2A FF EE 7D D6 80 23 20 21 00 00 22 FF EE C9 1C6
D170 2A 01 EF 23 4E 26 00 56 45 EE C9 2A FB EE 11 01 1DE
D180 00 CD C0 15 7D B4 28 07 21 00 00 22 07 EF C9 2A 12E
D190 0B EF 11 FC FF 19 22 09 EF 2A 0D EF 11 FC FF 19 184
D1A0 7E 23 66 6F EB 2A 0F EF 87 ED 32 38 0A 2A 0D EF 1E7
D1B0 11 FB FF 19 22 09 EF 2A 0F EF 11 04 00 19 22 09 186
D1C0 EF 7E 23 66 6F ED 5B 0F EF 87 ED 52 20 0D 2A 09 101
D1D0 EF 23 23 7E 23 66 6F 22 07 EF C9 2A 09 EF 23 7E 14F
D1E0 FE FF 20 D1 01 00 22 FD EE C3 EE CD 26 16 1FC
D1F0 22 11 EF 21 00 00 22 13 EF 2A 05 EF ED 5B 11 EF 1CD

```

```

sums 53 1C 28 52 98 41 DB 11 00 EB A9 35 9F CA 7A 173
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D200 07 ED 52 7D D6 03 BA 20 18 2A 11 EF 4E 26 00 22 11B
D210 FF EE 2A FF EE 7D D6 21 B4 28 10 2A FF EE 7D D6 1CE
D220 2A BA 28 2D 0B D2 02 ED EB 1C 2A 11 EF 28 36 18F
D230 11 2A 11 EF 25 2A 05 EF 28 2B 7E 23 66 6F 1E 1A6
D240 73 23 72 2A 05 EF 28 22 05 EF 21 01 00 22 13 EF 1AD
D250 C9 2A 11 EF 28 36 ED 2A 11 FB 3E 5B 2A 05 EF 1E3
D260 ED 5B 15 EF 87 ED 52 11 03 00 CD D5 15 E5 2A 17 133
D270 EF ED 5B 19 EF CD 05 15 EB 1C CD 15 7D B4 CD 155
D280 CD 9B D2 01 2A 18 05 CD 9B D2 01 22 2A 19 EF 1B2
D290 2A 05 EF 73 73 23 CD 22 05 EF C9 E3 4E 23 06 00 1FC
D2A0 ED 5B 05 EF ED 80 ED 53 05 EF C3 C9 2A F1 EE 22 1E4
D2B0 01 2A F3 EE 22 05 ED 2A F5 EE 11 78 00 22 1E2 1E2
D2C0 0B EF 22 0D EF 22 1B EF 21 00 20 22 10 EF 22 1F 1F4
D2D0 EF 2A 01 EF 4E 26 00 56 2E 00 E5 2A 01 EF 23 4E 1C0
D2E0 2A 00 EB 11 19 22 21 EF 2A 01 EF 7E FF 28 37 131
D2F0 21 00 00 22 23 EF CD 7A EC 2A 0D EF ED 5B 21 EF 106

```

```

sums 2F 51 A6 0E 0D D9 C3 92 1D ED F5 EF FA B5 13 A6 195
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D300 73 23 72 2A 0D EF 23 23 ED 5B 05 EF 73 73 72 2A 1E2
D310 0D EF 11 04 00 19 22 0D EF CD 96 D3 21 01 00 22 1C2
D320 23 EF CD 7A EC 18 AA 2A 1B EF ED 5B 08 EF 87 ED 121
D330 52 28 19 CD 09 15 CD 1E 15 4E 45 5B 5A 20 6E 6F 189
D340 74 20 66 6F 75 4E 44 21 00 C3 A8 EE CD 0D EF 11 124
D350 FF 73 23 72 2A 0D EF 23 23 ED 5B 05 EF 73 73 144
D360 72 2A 0D EF 11 04 00 19 22 0D EF CD 9B D2 01 C3 1E2
D370 2A EB EE 11 06 00 19 EB 2A 05 EF 73 73 72 23 22 189
D380 05 EF 2A FB EE 11 01 00 CD 15 7D B4 28 06 2A 1B4
D390 05 EF 22 25 EF C9 2A 01 EF 23 23 22 0F 21 189
D3A0 00 00 22 15 EF 21 00 00 22 27 EF 2A 01 EF 7E FE 115

```

```

D3B0 0D 20 0B 2A 01 EF 23 22 01 EF C9 CD C0 D3 1B EB 1B0
D3C0 CD 27 D1 2A FF EE 7D D6 3A B4 20 0A 2A 01 EF 23 184
D3D0 22 01 EF CD 27 D1 2A FF EE 7D D6 80 B4 CA 7B D5 18F
D3E0 CD 27 D1 2A FF EE 7D D6 5B B4 CA 80 B4 2A FF EE 1A3
D3F0 7D D6 21 B4 CA 56 EB CD 5E 59 CD 27 D1 2A FF EE 123

```

```

sums 54 B0 65 3B BC BE A3 27 3B 24 B0 F6 FA 7D 10 C9 1DD
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D400 7D D6 5B B4 28 3A 2A FF EE 7D D6 84 B4 28 5B CD 151
D410 5D 01 2A FF EE 7D D6 B4 C2 C3 2E 2A 19 EF CD 1A2
D420 14 16 CD 26 E1 CD 26 16 22 19 EF 2A 05 EF 22 15 186
D430 EF 2A 19 EF 22 17 EF C3 87 D2 2A 01 EF 23 22 01 C3
D440 EF CD 5D D2 CD 9F D4 CD 80 E6 CD B4 D4 CD 9B D2 18D
D450 01 73 2A 13 EF 7D D6 01 84 2A 0A 05 EF 28 22 1FE
D460 05 EF 28 36 36 C9 2A 01 EF 23 22 01 EF CD 50 1F8
D470 D2 CD D2 E5 CD 7F D4 CD 9B D2 03 73 23 7C CD 151
D480 D2 EB 18 06 CD 27 D1 CD 50 E6 CD 5D D1 2A FF EE 180
D490 73 D6 B4 C2 C3 EE 2A 01 EF 23 23 2A E1 C3 ED D1 1CD
D4A0 9B D2 01 E5 2A 05 EF CD 14 16 CD 3A E1 C3 ED D1 1CD
D4B0 2A 01 EF 23 22 01 EF CD 5E 9F CD 5D D1 2A FF EE 175
D4C0 7D D6 89 B4 C2 C3 EE 2A 01 EF 23 23 22 01 EF CD 1A2
D4D0 27 D1 2A FF EE 7D D6 5D B4 28 21 CD 5D D1 2A FF EE 1E0
D4E0 EE 7D D6 89 B4 C2 C3 EE 2A 01 EF 23 23 22 01 EF 1C3
D4F0 CD 4D D5 CD 8B D2 EA 02 72 23 C3 2B D4 CD 4D D5 2A 19B

```

```

sums 22 E5 D9 8B B2 EA E2 99 E0 14 07 57 CD 05 B0 2D 1BC
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D500 05 EF 11 F7 FF 19 7E 7E F1 C2 2B 2A 0A 05 EF 11 A1
D510 F7 FF 19 22 05 EF 23 6E 26 00 22 29 EF CD 5D D2 112
D520 2A 05 EF 28 28 7E FE 22 20 09 2A 05 EF 28 2B 1A4
D530 28 22 05 EF CD 9B D2 01 36 2A 29 EF EB 2A 05 EF 1FD
D540 73 23 22 05 EF CD 9B D2 01 C3 C3 2B D4 CD 50 E6 1CF
D550 CD 5D 01 2A FF EE 7D D6 B4 C2 C3 EE 2A 19 EF 1A2
D560 CD 14 16 CD 26 E1 CD 9B D2 01 EB CD 26 16 22 19 135
D570 EF CD 8D D2 CD 9B D2 02 73 23 2A 01 EF 23 22 10B
D580 01 EF 7E 23 22 01 EF 6F 26 00 22 FF EE 2A FF EE 15E
D590 11 20 00 CD EF 15 7D B4 C2 C3 EE 2A FF EE 11 30 157
D5A0 00 CD EB 15 7D B4 C2 2A FF EE 11 D1 FF 19 7C 87 1FF
D5B0 20 20 2D CA B6 D6 2D CA B5 D6 2D CA B6 D8 2D CA 191
D5C0 77 D9 2D CA E1 D1 2A D6 2A 66 D6 2D CA 75 D4 2A 1E6
D5D0 19 D8 2A FF EE 11 C9 FF 19 7C 87 20 2D 2A 21 186
D5E0 CD 2D CA 77 CD 2D CA DC 2D CA 10 DC 2D CA AB 184
D5F0 DD 2D CA B2 D6 2D CA 10 DC 2D CA 10 DC 2A FF EE 13D

```

```

sums C7 B0 25 72 0B EA BB CD 3E 9E 7E 56 B3 AE A3 2D 1AC
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D600 11 C1 FF 19 7C 87 20 2D 2D CA 10 2D CA 05 EF 11 A1
D610 2D CA 22 DE 2D CA 7C DE 2D CA 9C DE 2D CA BF DE 140
D620 2D CA D3 DE 2D CA DF DE 2A FF EE 11 89 FF 19 7C D1
D630 87 20 18 2D CA 2F EB D2 CA 41 EB D2 CA 2D DC 2D 153
D640 28 70 2D 28 6D 2D 28 6A 2D 28 67 2D 28 64 2A FF 187
D650 EE 11 B1 FF 19 7C 87 20 2F 2D 28 5A 2D CA EB DE 1A2
D660 2D CA F7 DE 2D CA 02 DF 2D CA 14 DF 2D CA 23 DF 187
D670 2D CA 32 DF 2D CA 41 DF 2A FF EE 11 A9 FF 19 7C 184
D680 87 20 2D CA 01 DF 2D CA 5B DF 2D CA 5F DF 2D 1AA
D690 CA 2F DF 2D CA 7F DF 2D CA 8F DF 2D CA 94 DF 2D 160
D6A0 FF 0E 2A FF EE 11 A1 FF 19 7C 87 20 04 2D CA E4
D6B0 19 E0 C3 23 EE C9 CD 3A E1 21 00 00 22 28 EF 21 1F6
D6C0 C2 00 22 2D EF CD FE D7 2A 28 EF 7D B4 CA 62 D7 11A
D6D0 2A 2B EF 7D D6 01 B4 20 13 2A 05 EF 28 2B 28 22 140
D6E0 05 EF CD 9B D2 03 87 ED 52 C3 AC 2F 2A 05 EF 11 10C
D6F0 FA FF 19 22 05 EF 23 6E 26 00 22 FF EE 2A 05 EF 15C

```

```

sums E1 1C AF FA 7D FB B0 D2 1A 25 D2 C0 D5 FD 7B DB 1AB
adrs +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F sum
D700 ED 5B 27 EF 87 ED 52 7D D6 03 B4 20 3A 2A 05 EF 1D2
D710 2B 28 22 05 EF 2D 2F 7D B4 28 05 CD 9B D2 177
D720 01 7E 2A FF EE 7D B4 20 07 CD 9B D2 01 87 18 3C 134

```


リスト 2 WICSコンパイラ・オブジェクト

EBE0	22	01	EA	AF	26	22	FF	EE	70	D6	2E	B4	C2	23	EA	1B	AE	
EBF0	CD	79	E4	CD	7B	D1	2A	07	EF	ED	5B	ED	19	EB	2A	EA	1B	
SUM1	DB	9C	74	5F	6B	85	97	52	9A	40	17	61	6D	AB	62	7B	194	
adr8	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	sum	
EC00	05	EF	73	23	72	23	22	05	EF	C3	67	26	84	C2	01	5F	15F	
EC10	23	22	01	EF	6F	26	22	05	FF	EE	70	D6	2E	B4	C2	23	1F3	
EC20	EE	06	CD	3E	9F	BC	D5	C3	67	26	2A	01	EF	7E	23	22	1F3	
EC30	EE	06	CD	3E	9F	BC	D5	C3	67	26	2A	01	EF	7E	23	22	1F3	
EC40	FF	EE	70	D6	22	B4	CA	67	26	2A	01	EF	7E	23	22	05	15D	
EC50	73	23	22	05	EF	1B	D5	23	53	EF	21	17	00	05	21	04	13D	
EC60	00	6E	26	00	2B	EB	E1	63	22	03	00	1E	15	24	00	13	7	
EC70	2A	53	EF	CD	15	15	CD	09	15	C9	2A	7F	1E	70	B4	CD	1C	
EC80	2A	5B	EE	11	01	00	CD	15	70	B4	2A	7F	1E	70	B4	CD	1C	
EC90	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC00	E1	CD	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
EC10	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC20	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC30	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC40	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC50	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC60	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC70	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC80	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC90	04	00	CD	DE	15	E5	2A	7F	1E	70	B4	2A	7F	1E	70	B4	CD	
EC00	25	25	25	20	43	6F	40	19	15	CD	1E	15	4F	62	6A	65	63	
EC10	25	25	25	20	43	6F	40	19	15	CD	1E	15	4F	62	6A	65	63	
SUM1	FB	2B	0B	D3	EA	D9	4F	CD	9A	C4	63	9A	FB	B4	E2	3B	1F9	
adr0	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	sum	
EBC0	73	69	78	5F	67	26	2A	01	05	00	22	84	15	2A	25	EF	154	
ED10	ED	5B	ED	5B	ED	5B	ED	5B	ED	5B	ED	5B	ED	5B	ED	5B	ED	
ED20	2A	05	EF	2A	05	EF	2A	05	EF	2A	05	EF	2A	05	EF	2A	05	
ED30	2A	05	EF	2B	CD	15	15	CD	1E	15	29	00	CD	09	15	CD	14	
ED40	09	15	15	1E	15	20	20	25	25	25	20	56	61	72	69	61	1E0	
ED50	62	6C	63	73	20	25	25	25	00	09	15	2A	1D	1F	22	78	7	
ED60	49	EF	21	00	00	22	48	EF	2A	49	EF	5B	1F	4F	5B	136	106	
ED70	F2	15	70	B4	C2	13	EE	EE	EE	EE	1B	0C	2A	4B	EF	7B	1D6	
ED80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
ED90	ED	5B	EE	1F	CD	15	15	2A	25	EF	ED	5B	4B	EF	1E	100	100	
EDAO	19	11	00	00	73	23	72	2A	4B	EF	23	22	48	EF	CD	1E	100	
EDB0	15	2A	01	00	08	00	22	55	EF	ED	BC	ED	2A	55	EF	2B	ED	
EDCO	22	55	EF	2A	49	EF	7E	23	22	49	EF	6F	26	00	CD	131	1	
EDDO	15	2A	49	EF	6E	26	00	11	00	00	CD	D5	15	7C	85	E1	151	
EDEO	20	02	AE	9F	21	00	00	22	57	EF	2A	55	EF	22	39	EF	152	
EDFO	CD	1E	15	20	00	2A	57	EF	23	22	57	EF	2A	39	2A	5F	178	
SUM1	82	DC	64	42	1A	91	D9	5F	5F	5C	1E	5E	79	39	D1	8D	515	
adr5	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	sum	
EE00	B7	ED	5D	F2	F0	ED	2A	49	EF	23	22	49	EF	CD	1E	EE	178	
EE10	C3	6B	ED	5D	CD	09	15	CD	09	15	CD	06	15	21	11	00	CD	1D5
EE20	05	C9	EF	17	21	05	00	22	84	15	CD	09	15	CD	1E	1C7	177	
EE30	05	C9	EF	17	21	05	00	22	84	15	CD	09	15	CD	1E	1C7	177	
EE40	20	49	AE	70	45	AE	45	AE	45	78	25	15	72	15	1D	1D	1D	

[illegible]

リスト3 P1の計算プログラム(MAX万桁)

```

10 '----- PI -----//
20 ' 2-2-7 295#
30 '
40 'PI=241ARCTAN(1/8)+81ARCTAN(1/57)+41ARCTAN(1/239)
50 '
60 '
70 '
80 ' WRITTEN BY K.YAMAMOTO
90 '
100 PRINT /"----- PI -----//
110 INPUT " 2-2-7 295# ?":N
115 IF (N(2)+(N(1)2000) GOTO 110
117 PRF=0
120 PRINT "PRINTER (Y/N):";IF GET="Y" THEN PRF=1
122 PRINT /
125 WORD=(N-N/6)/4+2;BYTE=WORD*2;CBYTE=BYTE-1;TIME=0
130 "-- ARRAY DEFINITION --
140 TERM1=64000;TERM2=TERM1*BYTE+4;TERM3=TERM2*BYTE+4
150 SUM=TERM3*BYTE+4;PI=SUM*BYTE+4;PIB7N=PI*BYTE+4
160 "-- ARRAY CLEAR --
170 N=TERM1
180 REPEAT:M(0)=0;M=N+1;UNTIL M=PIB7N
190 "-- INITIAL VALUE --
191 ' TERM1(0)=24/8
192 ' TERM2(0)= 8/57
193 ' TERM3(0)= 4/239
200 TERM1(0)=3
210 TERM2(0)=8
220 DIVIDEND=TERM2;DIVISOR=57 :GOSUB 5000
230 TERM3(0)=4
240 DIVIDEND=TERM3;DIVISOR=239;GOSUB 5000
250 '
260 GOSUB 6000 : 'SUM(0)=TERM1
270 GOSUB 6200 : 'SUM(1)=SUM+TERM2
280 GOSUB 6300 : 'SUM(2)=SUM+TERM3

```

```

290   GOSUB 2020 :PI ==PI+SUM
300   '
310   TM2FLG:=1;TM3FLG:=1;SIGM=1
320   FOR COUNT=1 TO BYTE#4/3
330     GOSUB 1000;PRINT COUNT
340   NEXT
350   PRINT /*      8 BINARY TO DECIMAL CONVERTING... JUST MOMENT!*/
400   GOSUB 2000;TI=TIME: BIN -> DECIMAL
420   IF PRF=1 THEN PRMODE 2
440   GOSUB 500
460   END
500   /* PRINT OUT
520   SPC10=0;CRLF=0;VTAB=0
530   PRINT /-----
540   PRINT "          PI = 3, */
560   FOR J=1 TO M
580     PRINT & TERM1[J]
600     GOSUB 700
620   NEXT
630   PRINT /"-----
640   PRINT "          TIME: ",TI," SEC."/
650   RETURN
700   /* TAB
710   IF INC(SPC10)<>5 RETURN
720   SPC10=0;PRINT " "
740   IF INC(CRLF)<>10 RETURN
760   CRLF=0;PRINT /
780   IF INC(VTAB)<>5 RETURN
800   VTAB=0;PRINT /;RETURN
900   '
1000  '#####
1020  GOSUB 8000
1030  DIVIDEND=TERM2:DIVISOR=57457
1040  IF TM2FLG<>0 THEN GOSUB 5000;TM2FLG=ZFLG
1050  DIVIDEND=TERM3:DIVISOR=239

```

リスト3 P1の計算プログラム(MAX1万桁)

```

1060 IF TM3FL6<0 THEN GOSUB 5000:GOSUB 5000:TM3FL6=ZFL6
1070 GOSUB 6000          :SUK==TERM1
1080 IF TM2FL6 THEN GOSUB 6200
1090 IF TM3FL6 THEN GOSUB 6300
1100 DIVIDEND=SUM:DIVISOR=COUNT*2+1
1110 GOSUB 5000          :SUK==SUM/(COUNT*2+1)
1120 SIGN=-SIGN:GOSUB 7000  :P1(==PI+SIGN*SUM
1130 RETURN
1140
1150 *
2000 * P1(BINARY) ----> TERM1(DECIMAL)
2010 Q=0:LEN=BYTE
2020 FOR J=0 TO M*4
2030 TERM1(J)=PI(O)*0*:PI(O)=0
2040 Q=Q+1:IF Q=6 THEN Q=0:LEN=LEN+2
2050 CLEN=-LEN-1:GOSUB 7400
2060 NEXT
2070 RETURN
5000 *- DIVISION -
5002 ZFL6=1
5010 : (2A V.DIVIDEND)      :HL=(DIVIDEND)
5020 : (ED4B V.WORD)         :DE=(WORD)
5030 : (B9 21 0000)         :EXX HL=0
5040 : (ED5B V.DIVISOR D9)  :DE=(DIVISOR) EXX
5050 *SKIP ZERO
5060 : (56 23 5E)           :DE=(HL+)
5070 : (7A 83 C2 L.5100)    :IF DE<0 GOTO L.5100
5080 : (23 08 7B B1 C2 L.5050):HL+ IF DEC(BC)<0 GOTO L.5050
5090 ZFL6=0:RETURN
5100 * DIVL
5110 : (2B C3 L.5130)       :HL- GOTO L.5130
5120 : (56 23 5E 2B)       :DE=(HL)
5130 : (B5 99 C1)          :DE EXX JBC
5140 GOSUB 5200
5150 : (CS 99 D1)          :BC EXX JDE
5160 : (72 23 73 23)       :HL+=DE
5170 : (0B 78B1 C2 L.5120) :IF DEC(BC)<0 GOTO L.5120
5180 RETURN
5200 *DIVW HLBC/DE-->BC...HL
5210 : (3E04)
5220 : (CB21 CB10 ED6A OC ED52):HLBC=HLBC*2 HL=HL-DE
5225 : (3002 19 0D)          :IF CY THEN HL=HL+DE C+
5230 : (CB21 CB10 ED6A OC ED52)
5235 : (3002 19 0D)
5240 : (CB21 CB10 ED6A OC ED52)
5245 : (3002 19 0D)
5250 : (CB21 CB10 ED6A OC ED52)
5255 : (3002 19 0D)
5260 : (3D C2 L.5220 C9)     :IF DEC(1A)<0 GOTO L.5220
6000 * SUK==TERM1
6010 : (2A V.TERM1)         :HL=(TERM1)
6020 : (ED5B V.SUM)         :DE=(SUM)
6030 : (ED4B V.BYTE)        :BC=(BYTE)
6040 : (EDB0 C9)            :LDIR RET
6200 * SUK==SUM+TERM2
6210 : (2A V.TERM2)         :HL=(TERM2)
6220 GOTO 6320
6300 * SUK==SUM+TERM3
6310 : (2A V.TERM3)         :HL=(TERM3)
6320 : (ED5B V.SUM)         :DE=(SUM)
6330 : (ED4B V.BYTE)        :BC=(BYTE)
6340 : (09 EB 09 EB)        :HL=HL+BC DE=DE+BC

```

```

6350 : (ED4B V.CBYTE)      :BC=(CBYTE)
6360 : (AF)                :CY=0
6370 : (1A BE 12 1B 2B)    : (DE)=(DE)+. (HL) HL- DE-
6380 : (0C C2 L.6370)     :IF INC(C)<0 GOTO L.6370
6390 : (04 C2 L.6370)     :IF INC(B)<0 GOTO L.6370
6400 RETURN
7000 * P1(==PI+SIGN*SUM
7010 IF SIGN=-1 GOTO 7100
7020 : (2A V.SUM)          :HL=(SUM)
7040 : (ED5B V.P1)         :DE=(P1)
7050 GOTO 6330
7100 * P1(==PI-SUM
7110 : (2A V.SUM)          :HL=(SUM)
7120 : (ED5B V.P1)         :DE=(P1)
7130 : (ED4B V.BYTE)       :BC=(BYTE)
7140 : (09 EB 09 EB)       :HL=HL+BC DE=DE+BC
7150 : (ED4B V.CBYTE)     :BC=(CBYTE)
7160 : (AF)               :CY=0
7170 : (1A 9E 12 1B 2B)    : (DE)=(DE)-. (HL) HL- DE-
7180 : (0C C2 L.7170)     :IF INC(C)<0 GOTO L.7170
7190 : (04 C2 L.7170)     :IF INC(B)<0 GOTO L.7170
7200 RETURN
7400 * P1=PI*10
7404 : (2A V.P1)           :HL=(P1)
7408 : (ED5B V.SUM)        :DE=(SUM)
7410 : (ED4B V.LEN)        :BC=(LEN)
7420 : (EDB0)             :LDIR
7430 : (2A V.P1)           :HL=(P1)
7432 GOSUB 7500
7442 : (2A V.P1)          :HL=(P1)
7474 GOSUB 7500
7475 : (2A V.SUM)         :HL=(SUM)
7482 GOSUB 7500
7490 : (2A V.P1)          :HL=(P1)
7492 GOSUB 7500
7493 RETURN
7500 : (ED4B V.LEN)        :BC=(LEN)
7510 : (ED5B V.P1)         :DE=(P1)
7520 : (09 EB 09 EB)       :HL=HL+BC DE=DE+BC
7530 : (ED4B V.CLEN)       :BC=(CLEN)
7540 GOTO 6360
8000 *- TERM1(==TERM1/64
8020 TERM1BTH=TERM1*BYTE:TERM1BTH1=TERM1BTH-1
8030 : (ED4B V.BYTE)       :BC=(BYTE)
8040 : (ED5B V.TERM1BTH)   :DE=(TERM1BTH)
8050 : (2A V.TERM1BTH1)    :HL=(TERM1BTH)-1
8060 : (EDB0):TERM1(O)=0
8070 GOSUB 8100:GOSUB 8100
8090 RETURN
8100 * TERM1=TERM1*2
8110 : (ED4B V.CBYTE)      :BC=(CBYTE)
8120 : (2A V.TERM1BTH)     :HL=(TERM1BTH)
8130 : (AF)               :CY=0
8140 : (7E BF 77 2B)       :HL)= (HL):2:HL-
8150 : (0C C2 L.8140)     :IF INC(C)<0 GOTO L.8140
8160 : (04 C2 L.8140)     :IF INC(B)<0 GOTO L.8140
8170 RETURN

```

リスト4 実行例

----- P1 -----

7*19 1*7*27*7 :1000

P1 = 3. +

14159 26535 89793 23846 26433 83279 50288 41971 69399 37510
 58209 74944 59230 78164 06286 20899 86280 34825 34211 70679

82148 08651 32823 06647 09384 46095 50582 23172 53594 08128
 48111 74502 84102 70193 85211 05559 64462 29489 54930 38196
 44288 10975 66593 34461 28475 64823 37867 83165 27120 19091

45648 56692 34603 48610 45432 66482 13393 60726 02491 41273
 72458 70066 06315 58817 48815 20920 96282 92540 91715 36436
 78925 90360 01133 05305 48820 46652 13841 46951 94151 16094
 33057 27036 57595 91953 09218 61173 81932 61179 31051 18548
 07446 23799 62749 56735 18857 52724 89122 79381 83011 94912

リスト4 実行例

98336 73362 44065 66430 86021 39494 63952 24737 19070 21798
 60943 70277 05392 17176 29317 67523 84674 81846 76694 05132
 00056 81271 45263 56082 77857 71342 75778 96091 73637 17872
 14684 40901 22495 34301 46549 58537 10507 92279 68925 89235
 42019 95611 21290 21960 86403 44181 59813 62977 47713 09960

51870 72113 49999 99837 29780 49951 05973 17328 16096 31859
 50244 59455 34690 83026 42522 30825 33446 85035 26193 11881
 71010 00313 78387 52886 58753 32083 81420 61717 76691 47303

59825 34904 28755 46873 11595 62863 88235 37875 93751 95778
 18577 80532 17122 68066 13001 92787 66111 95909 21642 01989

TIME: 75 SEC.

エディタ・アセンブラ

1. 概 要

本プログラムはMZ-80Bによりザイログ形式のニモニックで記述されたソース・プログラムを編集、アセンブルするものです。

編集コマンドによりメモリ上にソース・プログラムを形成し、アセンブル・コマンドでメモリ上のソース・プログラムをアセンブルするので内部処理は2パスですが、実際の操作は1パスで行なえます。メモリに余裕があれば、ソース・プログラム、オブジェクト・プログラムをオンメモリでデバッグできるので、誤りがあった場合ソース・プログラムの修正、再アセンブルが直ちに行なえます。

ソース・プログラムをメモリに持つということは、メモリ容量によりアセンブルできるプログラムの規模が制限される欠点もありますが、52Kバイト弱の空メモリがあるので、4,500ステートメント程度のソース・プログラムまでアセンブルできます。

EDASの主な機能は以下のとおりです。

- ① キーボード入力によりソース・プログラム作成
- ② ソース・プログラム・テープの作成、読み込み
- ③ 任意ステートメントおよびストリング・サーチによる特定ステートメントの訂正
- ④ ステートメント単位の移送
- ⑤ ソース・プログラム・リスト、アセンブル・リスト、ラベル・テープ・リストの作成
- ⑥ オブジェクト・プログラム・テープの作成、読み込み
- ⑦ アセンブル時、オブジェクト・コードを直接メモリに格納
- ⑧ Z80CPUの全レジスタ値を表示するブレイク機能

2. アセンブリ言語

2.1 文字

20H～F7Hの特殊文字、数字、英字、カタカナ、グラフィックなど216種類の文字を使用します。コメント欄ではこのすべての文字が使えます。

① ニモニック、レジスタ名、コンディション

41H～5AHの英文字(A～Z)を使います。

② 数字

a) 2進数: 0, 1

b) 10進数: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

c) 16進数: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

③ ラベル

3FH～F7Hのいずれかの文字で始まり、2文字目以後は、これに30H～39Hの数字(0～9)を含んでもかまいません。1ステートメントに収まる範囲内であるならば、文字数に制限はありません。

④ 演算子

+(加算), -(減算), *(乗算), /(除算を行ない, 商を得る), %(除算を行ない, 余りを得る), !(論理和), &(論理積)

2.2 定数

定数はそれ自身で定まる値を持つもので数字や文字の列で表現されます。数字だけで構成される数字定数、文字で構成される文字定数とがあります。

① 2進定数

2進定数は2進数文字で構成され、最後に“B”を付加します。

例: 1011B

② 10進定数

10進定数は10進数文字で構成されます。

例: 4096

③ 16進定数

16進定数は、16進数文字で構成され、最後に“H”を付加します。A～Fの文字で始まる場合、ラベルと区別するために最初に“0”を付加します。

例: 1234H, 0F57H

④ 文字定数

文字定数は引用符(')で囲まれた文字列を値として持ちます。引用符を文字として確保するときは、引用符を2個続けます。

例: 'A B イロハ', 'XY' 'Z'

2.3 ステートメント

1ステートメントは1行79カラムに収まるように書いてください。ステートメントはラベル欄、コード欄、オペランド欄、コメント欄の4つに分かれています。フリーフォーマットなので各欄はどのカラムから始めてもかまいません。

何も記述しないヌルステートメント、ラベルだけのステートメント、コメントだけのステートメントなども記述できます。

● 1つ以上のオペランドがある場合

[▲ラベル▲:] ▲コード ▲オペランド▲ [(, ▲オペランド)] ▲;
コメント] **CR**

● オペランドがない場合

[▲ラベル▲:] ▲コード▲ [; コメント] **CR**

● EQUステートメント

▲ラベル▲EQU▲オペランド▲ [; コメント] **CR**

[]内の記述は任意、▲の位置にスペースがあってもよい、△の位置に1個以上のスペースが必要、コロンの(;)はラベルとコードの区切りとして使う、カンマ(,)はオペランドの区切りとして使う、セミコロン(;)はコメントの始まりを表す。

2.3.1 ラベル欄

ラベルは命令で参照するため各ステートメントにつける名前で、必要により記述しラベルの後にコロンの(;)を付けます。擬似命令であるEQUステートメントの場合ラベルは省略できません、この場合ラベルの後はコロンのなくスペースを入れます。

命令、擬似命令、レジスタ名はラベルとして使えません。

2.3.2 コード欄

コード欄には実行すべきマシン命令、擬似命令を記述します。各命令はザイログ社様のニモニックです。

2.3.3 オペランド欄

コード欄に記述された命令の要求するレジスタ名、コンディション、データなどのオペランドを書きます。複数のオペランドがある場合コンマ(,)で区切ります。オペランド・データは表現式により記述します。

①表現式

表現式は数値定数(2進、10進、16進)、文字定数、ラベル、*カレント・プログラム・カウンタ(\$)などを演算子で結合したものです。演算は常に左から実行され演算子による優先順位はありません。また、()による優先順位の変更もできません。

※カレント・プログラム・カウンタ: 命令の割り付けられたアドレスを値とするもので"\$"で表す。

```
例  ORG 1000H
     C1 EQU 13
     LD HL, $+1100B-512*0FAH%ABC/C1!
     OAA00H & 1234H
```

(注1) ~部は演算子。

(注2) 表現式中の文字定数は後から2文字(16ビット、例では"BC")が演算対象となる。

(注3) 例は、LD HL, 0230Hと同等です。

レジスタ名、コンディション以外はすべて表現式で記述できます。

例 TDAT EQU 54

BIT TDAT & 7, A ; = BIT 6, A

RST TDAT & 38H ; = RST 30H

②相対ジャンプ命令

相対ジャンプ命令(JR, DJNZ)のジャンプ先を示すオペランドが以下の形式であれば、アセンブラが2番地の補正を行います。

JR ラベル名 OPt 数値定数

JR \$ OPt 数値定数 OPt: 演算子

ラベル名、\$を2つ以上含む場合および文字定数を含む表現式であると演算結果がそのままオペランド値となります。

```
例 ① { ORG 10H
      JR $+20H ; マシン語 181E (補正あり)
      }
      ② { ORG 10H
          JR $+ ' ' ; マシン語 1830(補正なし)
          ②の文字定数はASCIIコードで20Hなのでみかけ上
          ①と同じですが生成されるマシン語は異なる。
```

2.4 擬似命令

擬似命令はORG, EQU, DEFW, DEFB, DEFS, DEFM, SKIP, ENDの8種類があります。

①ORG (ORIGIN)

フォーマット [ラベル:] ORG データ

オペランドの値をアセンブル開始アドレスとして設定します。オペランドの記述にラベルが含まれている場合は、そのラベルは既に定義されたものでなければなりません。

本命令を省略すれば0番地をアセンブル開始アドレスとします。

②EQU (EQUATE)

フォーマット ラベル EQU データ

オペランドの値をラベルに割り付けます。オペランドの記述にラベルが含まれている場合は、そのラベルはすでに定義されたものでなければなりません。

③DEFW (DEFINE WORD)

フォーマット [ラベル:] DEFW データ, データ...

オペランド値を16ビット・データとして、下位8ビット上位8ビットの順にメモリに割り付けます。コンマ(,)で区切り、任意数のオペランドが記述できます。

④DEFB (DEFINE BYTE)

フォーマット [ラベル:] DEFB データ, データ...

オペランド値を8ビット・データとしてメモリに割り付けます。コンマ(,)で区切り任意数のオペランドが記述できます。

⑤DEFS (DEFINE STORAGE)

フォーマット [ラベル:] DEFS データ

オペランドで指定されたバイト数のメモリを確保します。オペランドの記述にラベルが含まれている場合は、そのラベルはすでに定義されたものでなければなりません。

⑥DEFM (DEFINE MEMORY)

フォーマット [ラベル:] DEFM '文字列'

オペランドに記述された文字定数(引用符で囲まれた文字列)をメモリに割り付けます。オペランドに文字定数以外は記述できません。

```
例  DFFM 'ABCD' ; 4124344
     DEFB 'A', 'B', 'C', 'D' ;
     DEFW 'BA', 'DC' ;
```

上記の3つのステートメントはすべて同一のコードをメモリに割り付ける。

⑦SKIP

フォーマット [ラベル:] SKIP [データ]

アセンブル・リスト出力時、改ページを行ない見出しを印字します。オペランドの記述はオブジェクト・コード生成に影響を与えませんが、未定義ラベルを記述すればラベル・エラーとなり、コンマ(,)で区切り3つ以上のデータを記述すればSyntax Errとなります。文字定数によりコメントを記述してください。

例 SKIP 'Main Program'

⑧END

フォーマット [ラベル:] END [データ]

一般にはアセンブラにアSEMBル終了を指示する命令ですが、EDASは指定ステートメントをアSEMBルする方式なので無意味な命令です。

オペランドの記述に関してはSKIP命令と同じです。

例 END 'Program -1 END'

3. 操 作 法

EDASプログラムの操作法について述べます。

3.1 起動

EDASのコールド・スタート・アドレスは12A3H番地なので、プログラム・ロード後このアドレスから起動してきます。

EDASがコールド・スタートすると、画面をクリア後下記のメッセージが出力されます（一部は出力されない）。

Z-80 EDAS For MZ-80B Ver.1.1
Buffer Area-3500, FFBF [CR]

ソース・プログラム、ラベル・テーブル、オブジェクト・コードなどを格納するためのバッファ・エリアを指定します。開始アドレス、終了アドレスの順に16進数で入力します。16進数以外を入力すると再度“Buffer Area-”のメッセージが出力されます。また16進数は何桁入力しても下位4桁を有効データとして扱います。バッファ・エリアで指定された範囲外にソース・プログラム、ラベル・テーブルが格納されることはありません（オブジェクト・コードに関してはASコマンドの項参照）。

3500番地より前は、モニタ、EDASプログラムが格納されているので、指定しないでください。またFFFFHにスタック・ポインタが設定されるので、FFBFH番地以後も指定しないでください。

EDASのホット・スタート・アドレスは12A0H番地なので、モニタに戻った後に再起動する場合このアドレスより起動してください。

3.2 コマンド説明

EDASのコマンドは全部で20種あり、コマンド待ちのとき“>”が表示されます。コマンド・パラメータのあるコマンドを入力すると“:”が表示され、コマンド・パラメータの入力待ちとなります。コマンドおよびパラメータの入力途中で[BRAKE]キーを入力すればコマンド待ちになります。また、コマンド・エラー、パラメータ・エラーの場合“?”を表示後コマンド待ちとなります。

コマンドの実行は[CR]キーで開始されます。

3.2.1 EDAS制御コマンド

① モニタ起動 >N [CR]

SB-1520モニタをホット・スタートします。

② モード設定 >M [CR]

EDASの実行モードを設定するものです。以下のメッセージに従って、[Y]または[N]を入力してモードを設定してください。Y/N以外（[CR]）のみの入力は次のモード設定にスキップ）を入力すればパラメータ・エラーとなりコマンド待ちに、[BRAKE]キーを入力すればコマンド待ちになります。これらの場合、設定されたモードは変わりません。

Printer Output?

ソース・プログラム・リスト、アSEMBル・リスト、ラベル・テーブル・リスト、オブジェクト・コード・リストをプリンタに出力する場合は[Y]を入力します。

Syntax Check?

ソース・ステートメントをキーボードまたはカセットテ

ープより入力時、文法チェックを行なう場合は[Y]を入力します。文法チェック・モードを設定しておくと、1ステートメント入力されるごとに文法チェックを行ない、誤りがあると“Syntax Err”のメッセージを出力します。キーボード入力時ならば、入力したステートメントが再表示され訂正が可能です。カセットテープよりの入力時はコマンド待ちとなります。[N]を入力すれば文法チェックは行なわれないので単なるエディタとしても使用できます。

40 Character Display?

CRT画面の表示文字数を設定します。[Y]を入力すれば40字モードに、[N]を入力すれば80字モードとなります。

Statement Number Display?

Lコマンド実行時、出力リストにステートメント番号を付加する場合、[Y]を入力します。

3.2.2 編集コマンド

コマンド形式 >コマンド:P₁, P₂, P₃ [CR]

本コマンドには、コマンド・パラメータの指定が必要です。パラメータP₁, P₂, P₃は10進数で入力し、処理ステートメントを指定します。また10進数のかわりに“.”, “/”の文字によりEDASの持つポイント値を参照できます。“.”は前回のコマンドで最後に処理したステートメント番号に1を加えた値で、“/”は入力済最終ステートメント番号の値です。またポイント±10進数の指定も可能です。

パラメータ数はコマンドにより異なりますが、余分なパラメータは無視されます。ただし、4つ以上パラメータを入力すればパラメータ・エラーとなります。入力済みステートメント数より大きい値など不合理なパラメータを入力すれば同様にパラメータ・エラーとなります。

以後のコマンド説明中〔 〕はパラメータが省略できることを示しています。

① キーボード入力 >I: (P₁) [CR]

パラメータで指定する位置に、キーボードより入力されるステートメントを順次挿入します。パラメータを省略すれば最終ステートメントの次より入力されます（ステートメントが未入力であれば1ラインより入力開始）。

コマンドの終了は[BRAKE]キー入力か、バッファがいっぱいになり“Buffer Full”のメッセージが出力された場合です。

キーボード入力時、次の編集キーが使用できます。

→	カーソルを右に進める。ただし最終文字以後には進まない。
←	カーソルを左に進める。ただし第1文字以前には進まない。
INS	カーソル位置より右の文字を順繰りに右に進め、カーソル位置にスペースを入力する。
DEL	カーソル位置より1つ左の文字を削除しカーソルより右の文字を順繰りに左に進める。
TAB	カーソル位置より8の倍数カラムとなるまでスペースを入力する。
↓	カーソル位置より右の文字をすべて削除し、1ステートメントの入力を終了させる。
CR	1ステートメントの入力を終了させる。

② ソース・プログラム・テープ読み込み>R: (P₁) [CR]

パラメータで指定する位置にカセットテープにより読み込まれるステートメントを順次挿入します。パラメータを省略すれば最終ステートメントの次より入力されます（ステートメントが未入力であれば1ラインより入力開始）。読み込みの一時停止は[SPACE]キー入力（CRTにステートメントが表示されつつあるときのみ有効）で行なわれ、

再度のスペース・キー入力で読み込みを再開します。

コマンドの終了は **[BRAKE]** キー入力 (このときMT動作中であれば“MT Read Err”のメッセージ出力)、ソース・プログラム・テープ読み込み終了、バッファがいっぱいになった (“Buffer Full”のメッセージ出力)、カセットテープのリード・エラー発生 (“MT Read Err”のメッセージ出力) および Syntax Checkモード指定中の Syntax Err 検出などの場合です。

③訂正 >C: {P₁} **[CR]**

パラメータで指定するステートメントの訂正を行います。コマンドを投入すると訂正対象ステートメントが表示されますので編集キーにより訂正を行ってください。**[CR]** または **[↓]** キー入力で訂正は有効となり次のステートメントが表示され訂正対象ステートメントとなります。訂正の必要のないステートメントであれば **[CR]** または **[↓]** キーのみ入力してください。

パラメータを省略すれば、1ラインより訂正可能となります。本コマンド実行中も Syntax Checkは可能です。

コマンドの終了は、**[BRAKE]** キー入力、最終ステートメントの訂正終了、およびバッファがいっぱいになった (“Buffer Full”のメッセージ出力) ときなどの場合です。

※編集キーについては“II”コマンドの項参照

④削除 >D: {P₁ [, P₂]} **[CR]**

P₁~P₂ステートメントを削除します。P₁のみ指定すればP₁ステートメントが削除されます。P₁, P₂共に省略すれば全ステートメントの削除指定となります。このとき“Delete All Line, OK? Y/N-”のメッセージが出力されますので削除しても良い場合は **[Y]** を、削除したくない場合は **[N]** を入力します (Y, N以外はパラメータ・エラー)。

コマンドの終了は **[BRAKE]** キー入力または削除処理終了時などです。

⑤サーチによる訂正 >S: {P₁ [, P₂]} **[CR]**: 文字列 **[CR]**

P₁~P₂ステートメントの範囲で、文字列で指定されたものと同一の文字列のあるステートメントを見つけ出し、訂正対象ステートメントとして表示します。訂正が終了すれば、次の訂正対象ステートメントのサーチが行なわれます。すなわち特定ステートメントのみ訂正するためのコマンドです。

P₁のみ指定すれば、P₁ステートメントがサーチ対象となり、全パラメータ省略すれば全ステートメントがサーチ対象となります。

サーチできるのは15文字以下の文字列であり、16文字以上の入力および **[TAB]** 以外 (注: 8の倍数カラムまで連続したスペースはバッファ内で“TAB”コードに変換されます) の特殊文字入力はパラメータ・エラーとなります。本コマンド実行中でも“Syntax Check”は可能です。

コマンドの終了は **[BRAKE]** キー入力および指定ステートメントまでのサーチ終了時です。

⑥ソース・プログラム・リスト出力>L: {P₁ [, P₂]} **[CR]**

P₁~P₂ステートメントを出力し、ソース・プログラム・リストを作成します。P₁のみ指定すればP₁ステートメントが出力され、P₁, P₂共に省略すれば全ステートメントが出力されます。

[SPACE] キー入力では出力は一時停止となり、再度の **[SP ACE]** キー入力では出力が再開されます。

コマンドの終了は **[BRAKE]** キー入力および指定ステートメントまでの出力終了時です。

⑦ソース・プログラム・テープ出力>W: {P₁ [, P₂]} **[CR]**

P₁~P₂ステートメントをカセットテープに出力し、ソー

ス・プログラム・テープを作成します。P₁のみ指定すればP₁ステートメントが出力され、P₁, P₂共に省略すれば全ステートメントが出力されます。

コマンドの終了は **[BRAKE]** キー入力および指定ステートメントまでの出力終了時です。

⑧テープ・ベリファイ >V: {P₁} **[CR]**

P₁ステートメント以後の内容とWコマンドで作成したソース・プログラム・テープのベリファイを行います。P₁を省略すれば1ステートメントからベリファイが実行されます。

ベリファイの結果一致していれば“Verify OK”を、不一致であれば“Verify Err”のメッセージを出力します。

コマンドの終了は **[BRAKE]** キー入力 (このときMT動作中であれば“MT Read Err”のメッセージ出力)、カセットテープのリード・エラー発生 (“MT Read Err”のメッセージ出力)、ベリファイ終了時などです。

⑨移送 >T: P₁, P₂, P₃ **[CR]**

P₁~P₂ステートメントをP₃ステートメントの直前に移送します。パラメータは (P₁ ≤ P₂) AND ((P₂ + 1) < P₃) または (P₁ ≤ P₂) AND ((P₁ - 1) > P₃) の条件を満足しないとパラメータ・エラーとなります。なお、P₃として最終ステートメント番号 + 1も指定できます。

コマンドの終了は移送終了時です。

3.2.3 アセンブル・コマンド

コマンド形式 >Af: P₁, P₂ **[CR]**

P₁~P₂ステートメントのアセンブルを行います。fはL, W, S, Tのいずれかの文字でアセンブル内容の指定を行います。

パラメータの指定法は3.2.2項の編集コマンドと同様なのでそちらを参照してください。P₁のみ指定すればP₁ステートメントを、P₁, P₂共に省略すれば全ステートメントがアセンブルされます。アセンブル対象ステートメントの範囲外のラベルを参照するステートメントがあると未定義ラベル・エラーとなります。

コマンドを投入すると“EXC/Label Table GN”のメッセージを出力し、ラベル・テーブルが作成開始されます。ラベル・テーブル作成中ラベルに値が正しく割り付けられないステートメントがあると“Err in XXXX”のメッセージを出力し、コマンドの実行を終了します。なお、メッセージ中のXXXXは誤りのあるステートメント番号を表示します。

Syntax Checkモードを指定してソース・プログラムの入力を行えば大部分の誤りは入力時に発見できます。ただしORG, EQU, DEFS ステートメントのオペランドに記述された未定義ラベル・エラーはこのラベル・テーブル作成時に発見されエラーとなります。

二重定義ラベルのチェックはATコマンド実行時のみ行なわれます。その他のコマンド実行時はライン番号の小さいステートメントのラベル値を充ててアセンブルが行なわれます。

ラベル・テーブルの作成が終了すると以下に述べる各アセンブル処理が実行されます。

①アセンブル・リスト出力>AL: {P₁ [, P₂]} **[CR]**

アセンブル・リストを出力します。リストには、下記のものが出力されます。

PAGE	XXXX	: リストのページ数
E		: エラーコード出力欄
STNR		: ステートメント番号出力欄
ADRS		: 命令のアドレス, EQUステートメント

方はパラメータ・エラーになります。本コマンドを投入しなければEDASのコールド・スタート時RST 30Hが設定されています。

②ブレーク・アドレス指定 >BS:Pa CR

ブレーク・ポイント・アドレスを指定します。Paは16進数で入力し、下位4桁が有効です。最大10箇所まで指定できます。

16進数以外を入力、0番地の設定はパラメータ・エラーとなります。

③ブレーク・アドレス取り消し >BR:Pa CR

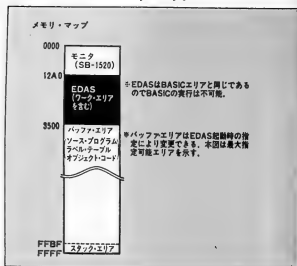
設定済みブレーク・ポイント・アドレスを取り消します。Paは16進数で入力し、下位4桁が有効です。

16進数以外を入力、未設定のアドレス指定の場合はパラメータ・エラーになります。

④ブレーク・ポイント・アドレス表示 >BD CR

設定済みブレーク・ポイント・アドレスをすべて表示します。

メモリ・マップ



4. 参 考

カセット・テープの記録方式は1,200BPSのカンサス方式です。下記のように変更すれば600BPSとなりますので、PC-8001用のEDASとソース・プログラム・テープ、オブジェクト・プログラム・テープの互換性があります。

アドレス	1200BPS	600BPS
2136H	01H	02H
215EH	0AH	02H
2160H	01H	03H
216CH	04H	08H
2203H	01H	02H
224DH	02H	04H

ALコマンド実行時のエラー表示例

PAGE	1			
E	STNR	ADRS	OBJECT	SOURCE STATEMENTS
	1			!Example for program
	2			!
	3			!シ"ヨウス"ノ タイキ"
	4	0FFF		CR EQU 0FFFF
	5	000A		LF EQU 0AH
	6	00B1		MONITOR EQU 0B1H ;Monitor entry
	7	16D7		PRINT EQU 16D7H
	8			!
	9			!Main program
	10	0000		ORG 9000H
L	11	9000	310000	LD SP,STACK
	12	9003	CDAF90	CALL メモセイシ"
R	13	9006	0000	DEFB CR,LF
	14	900B	20454441	DEFM ' EDASハ" サ"イロク" ケイシキノ Z80 ニイニツクヲ '
		900C	53CA20BB	
		9010	DEB2DBBB	
		9014	DE20B9B2	
		9018	8CB7C920	
		901C	5A383020	
		9020	C6B2D3C6	
		9024	AFBBA620	
	15	902B	B1AFBEDD	DEFM 'アッセンブ"ルシマス。'
		902C	CCDED9BC	
		9030	CFBDA1	
R	16	9033	0000	DEFB CR,LF
	17	9035	20CEB3CC	DEFM 'ホウフナ" ハンシュウキノクラ モリ エデ"イダツキデ"スノデ" マシンコ" '

```

9039 C520CDDD
903D BCADB3B7
9041 C9B3A620
9045 D3C220B4
9049 C3DEA8C0
904D C2B7C3DE
9051 BDC9C3DE
9055 20CFBCDD
9059 BADE
18 905B CCDFDBBB      DEFM      'プログラム/ カイハツカ' ヨウニ テキマス。'
905F DED7D1C9
9063 20B6B2CA
9067 C2B6DE20
906B D6B3B2C6
906F 20C3DEB7
9073 CFBDA1
R 19 9076 00000000      DEFB      CR,LF,LF,0
20 907A C3B100          JP          MONITOR ;return to monitor
21
22      ;Stack area
23 907D              DEFS      50
24 90AF      STACK:
25
26      ;Sub-routine
27 90AF E3      メモリ: EX      (SP),HL
28 90B0 7E      LD          A,(HL)
29 90B1 23      INC         HL
V 30 90B2 CB47      BIT          B,A
31 90B4 2B05      JR          Z,MSGPE
32 90B6 CDF616      CALL       PRINT
R 33 90B9 1B00      JR          MSGPE+140
34 90BB E3      MSGPE: EX      (SP),HL
35 90BC C9      RET

```

Lコマンド (ソース・プログラム・リスト)

ERRORS 6

```

1 ;Example for program
2 ;
3 ;シ"ヨウスウ / テキ"
4 CR      EQU      0DH
5 LF      EQU      0AH
6 MONITOR EQU      0BH      ;Monitor entry
7 PRINT   EQU      16D7H
8 ;
9 ;Main program
10      ORG      9000H
11      LD      SP,STACK
12      CALL    メモリ"
13      DEFB    CR,LF
14      DEFM    ' EDASハ サ"イロ" タイキノ ZBO ニイニヤクヲ '
15      DEFM    'アセンブ"ルシマス。'
16      DEFB    CR,LF
17      DEFM    ' ネット ハンショウキノヲ セツ イテ"イタツキチ"スノチ" マシンコ'
18      DEFM    'プログラム/ カイハツカ' ヨウニ テキマス。'
19      DEFB    CR,LF,LF,0
20      JP      MONITOR ;return to monitor

```

```

21 ;
22 ;Stack area
23      DEFS      50
24 STACK:
25 ;
26 ;Sub-routine
27 メッセージ: EX      (SP),HL
28      LD        A,(HL)
29      INC       HL
30      AND       A
31      JR        Z,MSGPE
32      CALL      PRINT
33      JR        メッセージ+1
34 MSGPE: EX      (SP),HL
35      RET

```

ATコマンド (シンボル・テーブル・リスト)

SYMBOL	ADRS	SYMBOL	ADRS	SYMBOL	ADRS
CR	000D	LF	000A	MONITO	00B1
MSGPE	90BA	PRINT	16F6	STACK	90AF
メッセージ	90AF				

AWコマンド (オブジェクト・コード・リスト, 同一内容がカセットテープに出力される)

```

1090000031AF90CDAF900D0A2045444153CA20BBEB
10901000DEB2DBBBDE20B9B2BCB7C9205A38302086
10902000C6B2D3C6AFBBA620B1AFBEDDCCDED9BCCB
10903000CFBDA10D0A20CEB3CCC520CDDDBCADB3D4
10904000B7C9B3A620D3C220B4C3DEABCC2B7C379
10905000DEBDC9C3DE20CFBCDDBADECCDFDBBBDECF
10906000D7D1C920B6B2CAC2B6DE20D6B3B2C620A6
0D907000C3DEB7CFBDA10D0A0A00C3B100D9
0190AF00E3DD
0C90B0007E23A72805CDF61618F6E3C9AC

```

ERRORS 0

ALコマンド (アセンブル・リスト)

PAGE	1				
E	STNR	ADRS	OBJECT	SOURCE	STATEMENTS
	1			;Example for program	
	2			;	
	3			;シ"ヨウスノ テイ"	
	4	000D		CR	EQU 00DH
	5	000A		LF	EQU 00AH
	6	00B1		MONITOR	EQU 00BH ;Monitor entru
	7	16F6		PRINT	EQU 16D7H
	8			;	
	9			;Main program	
	10	0000		ORG	9000H
	11	9000	31AF90	LD	SP,STACK
	12	9003	CDAF90	CALL	メッセージ
	13	9006	0D0A	DEFB	CR,LF
	14	9008	20454441	DEFM	' EDAS\ ヲ イロク\ ケイシノ ZBO ニイニツクラ '
		900C	53CA20BB		
		9010	DEB2DBBB		
		9014	DE20B9B2		
		9018	BCB7C920		
		901C	5A383020		
		9020	C6B2D3C6		
		9024	AFBBA620		
	15	9028	B1AFBEDD	DEFM	'アセンブ' &シマス.'
		902C	CCDED9BC		
		9030	CFBDA1		
	16	9033	0D0A	DEFB	CR,LF
	17	9035	20CEB3CC	DEFM	' ネットワ ハンシュウキノウラ モツ エテ'イダリキテ'スノテ' マシンコ' '
		9039	C520CDDD		
		903D	BCADB3B7		
		9041	C9B3A620		
		9045	D3C220B4		
		9049	C3DEABCO		
		904D	C2B7C3DE		
		9051	8DC9C3DE		
		9055	20CFBCDD		
		9059	BADE		
	18	905B	CCDFDBBB	DEFM	'フ'ロク'ラム' カイバツカ' ヨフニ テ'キマス.'
		905F	DED7D1C9		
		9063	20B6B2CA		
		9067	C2B6DE20		

```

906B D6B3B2C6
906F 20C3DEB7
9073 CFBDA1
19 907A 0D0A0A00      DEFB      CR,LF,LF,0
20 907A C3B100         JP          MONITOR ;return to monitor
21                      ;
22                      ;Stack area
23 907D                DEFS      50
24 90AF                STACK:
25                      ;
26                      ;Sub-routine
27 90AF E3             メモリ*1 EX      (SP),HL
28 90B0 7E             LD        A,(HL)
29 90B1 23             INC       HL
30 90B2 A7             AND       A
31 90B3 2B05           JR        Z,MSGPE
32 90B5 CDF616         CALL     PRINT
33 90B8 18F6           JR        メモリ*1 +1
34 90BA E3             MSGPE:    EX      (SP),HL
35 90BC C9             RET

```

ERRORS 0

実行結果

EDASは「アイロフ」 クイシキ/ Z80 ニモニツクワ アッセンブ「ルシマス」。
 ホワフ「ンシユウキ/ワラ モリ エチ「イタリキチ」/スノチ マシシコ「フ」ロウ「ラム」 カイハツカ」 ヨフイニ テ「キマス」。

EDASマシン語ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F :Sum
12A0 C3 08 17 31 FF FF 21 30 33 06 FF 36 00 23 10 FB :FD
12B0 CD EE 0C 3E 05 32 33 3E F7 32 D5 35 CD 15 17 :12
12C0 06 5A 2B 38 40 20 55 44 41 53 20 46 6F 72 20 4D :E6
12D0 5A 2B 38 40 20 55 44 41 53 20 46 6F 72 20 4D :E6
12E0 CD 15 17 0D 42 75 66 66 55 72 20 41 72 65 61 2D :26
12F0 00 CD B1 22 FE 2C 20 EB 22 4F 33 22 51 33 CD B1 :9A
1300 22 FE 0D 0B 22 53 33 CD 0F 17 3F FF 4F 32 :03
1310 39 33 CD 15 17 0D 95 17 2F 13 0A 0E :B8
1320 00 CD B1 20 34 21 3D 13 09 09 5E 23 56 EB 49 41 :61
1330 42 43 44 49 4C 4D 4F 52 53 54 56 57 4E 61 13 :67
1340 13 44 24 90 13 67 14 CA 14 70 15 7C 15 26 16 4B :EE
1350 16 4E 16 9F 22 EC 2E 15 7C 15 26 16 4B 1C :11
1360 AA CD 21 19 63 B1 00 00 CD D0 18 CD 52 19 AF 37 :D0
1370 8F 21 38 22 23 7E FE 1B 28 1F FE FA 3F 30 F0 3E :A0
1380 0D A7 CD 8F 21 38 02 0F CD 66 19 3B EB 06 0A 0F :66
1390 8F 21 38 02 10 1F CD 08 13 3E 09 1B D2 21 59 :13
Sum 5B 2B B3 AC 66 4F 6C 59 47 0C DC 90 B7 EE 71 73 :A4

```

```

15A0 33 CD 15 17 0D 53 79 4E 74 61 7B 20 43 6B 65 63 :13
15B0 6B 20 3F 0D 59 2F 4E 2D 00 CD 7D 19 3B 03 32 31 :D8
15C0 33 CD 15 17 0D 34 30 20 43 6B 61 72 61 63 74 65 :D8
15D0 72 20 43 6B 73 70 4C 61 79 20 3F 0D 59 2F 4E :D7
15E0 00 CD 7D 19 3B 11 20 07 CD EE 0C 3E 03 1B 05 CD :C5
15F0 7C 0C 3E 06 32 30 33 CD 15 17 0D 53 74 61 74 65 :17
1600 6D 65 6E 74 20 4E 75 6D 62 65 72 20 44 69 73 :ED
1610 6C 61 79 20 3F 0D 59 2F 4E 2D 00 CD 7D 19 3B :03
1620 32 32 33 CD 0B 13 CD 0D 1B CD 52 19 3E 4C 32 39 :5A
1630 33 CD 0F 17 E5 3A 32 33 AF CC F6 17 E1 CD E7 1A :09
1640 CD 66 19 3B EC CD 0F 17 C3 0B 13 CD 0A 18 22 49 :13
1650 33 CD 0C 1A CD 0F 17 E5 CD F6 17 E1 CD 99 19 DA :C6
1660 0B 13 CD E3 1B CD 33 1A 2A 49 33 23 1B E0 CD D0 :21
1670 1B 30 26 CD 15 17 0D 44 65 6C 74 65 20 41 6C :9A
1680 6C 2D 4C 69 6E 65 2C 20 4F 4B 3F 20 59 2F 4E :D9
1690 00 CD 7D 19 DA 59 13 20 03 CD 56 1B C3 0B 13 CD :B8
Sum 8C DB 26 B5 D0 9A EB 29 F2 B4 BF D3 92 FC 40 77 :13A

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F :Sum
13A0 E5 2B 38 40 20 55 44 41 53 20 46 6F 72 20 4D :E6
13B0 CD B1 22 FE 2C 20 EB 22 4F 33 22 51 33 CD B1 :9A
13C0 B9 CD 09 D1 23 CD 9A 17 0B 09 E5 D5 D9 C1 E1 :A1
13D0 EB CD 9A 17 38 4E EB 6D 69 23 CD D0 0D EB 1B :0E
13E0 A7 ED 52 E5 EB D9 E5 D9 D1 A7 ED 52 D1 CD 9A 17 :4D
13F0 D9 38 4E 05 D5 C1 D1 1B E1 23 C1 E5 EB A7 ED 52 :23
1400 27 E5 C1 32 49 33 EB CD 1A E5 D9 E1 54 5D CD :24
1410 56 1B 21 77 33 CD FE 1A D9 23 17 EB 0B 7B B1 20 :6F
1420 E2 C3 0B 13 EB E5 A7 ED 52 E3 EB 6D 49 A7 ED 52 :F6
1430 D1 CD 9A 17 D9 38 4A E5 D5 C1 C1 03 D1 1B C5 :F3
1440 EB A7 ED 52 23 44 0E 1B 28 22 33 EB CD 0A 1A :D1
1450 E5 D9 E1 54 5D CD 56 1B 21 77 33 CD FE 1A D9 23 :28
1460 7B B1 20 E9 C3 0B 13 CD 0B 1B CD 15 17 20 3A :3B
1470 00 21 66 33 06 10 CD 7D 17 77 23 FE 1B 28 04 FE :0E
1480 20 3B 0D 0B 81 16 10 E5 C3 59 13 CD 2A 19 7B FE :A1
1490 10 CA 59 13 2A 49 33 CD 52 19 CD 0F 17 E5 23 E5 :04
Sum 94 4F 2B 1A 9A 39 E4 92 EF AB 51 B9 DC 23 B9 73 :3A

```

```

16A0 B2 1B ED 53 45 33 CD 52 19 CD 93 1B 3B FB C3 0B :13
16B0 13 F5 CD C6 0B 3A 30 33 A7 20 1A 3A 33 FE 4C :11
16C0 2B 0F FE 5A 2B 04 FE 57 20 0B F1 F5 FE 0D 02 :41
16D0 3E 0A CD 07 16 F1 C9 F5 3E 00 CD ED 16 F1 D3 FF :82
16E0 3E 80 03 FE 31 C1 CD 1E 16 AF D3 FE C9 C5 D5 :D8
16F0 1E 04 01 00 0D FE E6 0A BA 20 03 D1 C1 C9 0B :32
1700 7B B1 20 F1 10 0D EE 3E 01 32 30 33 D1 C1 C9 :02
1710 0B CD B1 16 C9 E3 7E 23 A7 2B 05 CD B1 16 1B :6A
1720 E3 C9 C5 D5 E5 CD 29 0C 22 03 0D CD 3E 0C 32 :0D
1730 00 32 0E 00 2A 03 00 3A 0E 00 FE 1F 20 0B 3A :44
1740 00 32 0E 00 CD 50 0C 1B 04 3E 1F 1B F4 06 40 :F9
1750 CD 0F 09 FE 1E 20 05 C1 10 F5 1B 0B FE 09 2B :03
1760 FE 0A 2B 04 FE 0C 20 05 CD 39 0A 1B EA F5 2A :03
1770 00 3A 0D 00 CD 50 0C F1 C1 E1 D1 C1 C9 D2 27 :64
1780 FE FB 0B AF C9 CD 7D 17 FE 2B 0B CD B1 16 C9 :B7
1790 5F D0 1A C9 E5 A7 ED 52 E1 C9 E5 CD A9 1F 21 :00
Sum 17 69 35 9B 22 51 CB B3 9A F4 60 87 FE A6 3D 72 :11C

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F :Sum
17A0 00 3E 11 E5 19 30 01 E3 E1 C5 CB 11 CB 10 CB :15
17B0 CB 1A F1 3D 20 ED AF CB 1C CB 1D EB E1 C9 F5 :C7
17C0 01 0A 00 CD C9 17 C1 F1 C9 E5 D9 C5 D5 01 00 :71
17D0 0A 09 3E 10 A7 CB 1A CB 1B 30 07 C5 D9 E1 09 :4D
17E0 44 D9 D9 A7 CB 18 19 CB 1A CB 1B 30 07 C5 D9 :E4
17F0 E1 D1 C1 D9 E1 C9 2A 49 33 E5 C5 44 2E 04 0E :DE
1800 11 0A 00 CD 9A 17 F5 2D 20 F6 2A 0E D1 7B A7 :F6
1810 0A CD 0C 1C 2D 20 F3 D1 7B C6 30 CD B1 16 C9 :1C
1820 2D 20 F6 CD 0C 1C D1 E1 C9 CD 4A 19 3B 39 ED :FC
1830 5B 49 C3 FE 2E 2B 0B 11 00 00 FE 2C 20 25 ED :58
1840 4D 33 CD B5 17 FE 2B 2B 05 FE 2D 20 13 37 E5 :D5
1850 0B AF 6D 1B 08 30 CD C9 A9 F1 E1 19 EB 0B E1 :A7
1860 FE 2C 2B CD 01 09 C9 CD 6D 1B 13 F1 00 00 :6A
1870 CD 6F 17 CD 44 19 0D BE 17 1B F4 3E 3A CD B1 :11
1880 16 21 00 04 00 00 CD 2A 1B EB 3F 0F CD 2A 1B :B8
1890 3B 0A D5 CD 2A 1B D2 59 13 42 4B D1 CD 0F 17 :C9
Sum 02 E7 5D 0C 1B 08 92 E2 6D E9 BE 89 EA 17 91 D5 :FA

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F :Sum
18A0 CD 7C 1B EB 2A 4D 33 23 7A B3 CB ED 52 DA 59 :13
Sum 02 E7 5D 0C 1B 08 92 E2 6D E9 BE 89 EA 17 91 D5 :FA

```

EDASマシン語ダンプ・リスト

1880	EB	C9	2A	4D	33	7C	B5	CA	59	13	ES	CD	7C	18	7C	B5	13C
18C0	20	01	23	EB	E1	CD	94	17	DA	59	13	ES	22	49	C3	C9	120
18D0	2A	4D	33	7C	B5	CA	59	13	ES	CD	7C	18	7C	B5	28	10	180
18E0	7A	B3	20	02	54	50	EB	CD	94	17	DA	59	13	EB	ES	ED	164
18F0	52	DA	59	13	E1	22	43	33	22	49	C3	ES	53	45	C3	C9	130
1900	25	DA	59	13	EB	CD	32	08	EB	CD	08	13	FE	20	11	11	110
1910	CD	32	08	13	FE	20	11	11	11	11	11	11	11	11	11	11	110
1920	C9	CD	32	08	13	FE	20	11	11	11	11	11	11	11	11	11	110
1930	18	0D	0C	3E	20	CD	B1	16	0D	79	E6	07	20	F5	C9	CD	157
1940	B1	16	0D	0C	3E	20	CD	B1	16	0D	79	E6	07	20	F5	C9	157
1950	A7	C9	C5	44	2A	51	C3	7E	25	FE	3A	22	49	C3	CD	78	1C2
1960	B1	16	0D	0C	3E	20	CD	B1	16	0D	79	E6	07	20	F5	C9	157
1970	05	19	ED	0F	1F	33	27	18	7C	B5	CA	59	13	ES	CD	7C	18
1980	FE	08	CA	08	13	FE	20	11	11	11	11	11	11	11	11	11	110
1990	28	05	DA	59	C2	59	C2	59	A7	C9	AF	32	36	33	11	9D	19
Sum	C9	D3	3E	1D	FB	AB	DB	25	91	25	56	FC	CD	45	1A	36	101

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
19A0	D5	CD	70	17	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F
19B0	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	15F
19C0	1B	CA	5E	16	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F
19D0	03	CD	08	05	0D	73	23	7B	CD	B1	16	0D	79	E6	07	20	159
19E0	3A	36	33	A7	C9	FE	08	2B	3A	FE	0D	20	33	FE	03	28	159
19F0	05	CD	B1	16	C9	79	FE	08	2B	3A	FE	0D	20	33	FE	03	111
1A00	04	CD	B1	16	C9	79	FE	08	2B	3A	FE	0D	20	33	FE	03	182
1A10	0F	19	23	7E	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	182
1A20	FF	19	23	7E	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	182
1A30	16	0D	FF	19	23	7E	FE	0D	20	33	FE	03	28	35	15F	15F	182
1A40	7E	28	77	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	15F	182
1A50	CD	B1	16	0D	FF	19	23	7E	FE	0D	20	33	FE	03	28	35	182
1A60	20	0C	3D	08	05	0D	73	23	7B	CD	B1	16	0D	79	E6	07	1AC
1A70	20	0C	3D	08	05	0D	73	23	7B	CD	B1	16	0D	79	E6	07	1AC
1A80	23	18	0F	54	50	EB	CD	94	17	DA	59	13	ES	CD	7C	18	15B
1A90	20	F4	E5	18	10	CD	BB	1A	33	CB	FE	27	26	12	16D	16D	15B
Sum	04	AA	C6	30	D7	A0	4F	29	B2	53	10	22	10	15	0F	9B	169

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
1AA0	FE	20	03	EB	38	3E	3A	FE	0D	20	33	FE	03	28	35	15F	159
1AB0	18	D5	CD	BB	1A	20	FB	09	A7	C9	17	13	CD	0E	0D	C9	117
1AC0	21	77	33	01	50	50	55	C5	36	0D	23	10	FB	E1	C1	C9	1F2
1AD0	E5	05	11	77	33	02	52	19	7E	12	23	13	7E	FE	FA	38	121
1AE0	F8	3E	00	12	E1	C1	09	0E	50	23	7E	FE	FA	0D	20	2F	193
1AF0	19	18	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	15F	15F	193
1B00	FF	7E	28	77	FE	0D	20	33	FE	03	28	35	15F	15F	15F	15F	193
1B10	EB	2A	53	37	47	ED	52	DA	59	13	ES	CD	7C	18	7C	B5	10A
1B20	33	23	ED	52	E1	2B	1C	ED	52	DA	59	13	ES	CD	7C	18	10A
1B30	EB	2A	53	37	47	ED	52	DA	59	13	ES	CD	7C	18	7C	B5	10A
1B40	18	0B	D1	C1	E1	0D	0B	EB	36	FF	22	4F	33	2A	4D	01	1E
1B50	33	23	ED	52	33	C5	44	2A	59	13	ES	CD	7C	18	7C	B5	10A
1B60	28	03	E5	13	ED	52	40	35	ED	52	19	EB	2A	53	37	47	10A
1B70	4F	37	ED	52	33	C5	44	2A	59	13	ES	CD	7C	18	7C	B5	10A
1B80	ED	52	4F	33	C9	0F	28	43	23	CD	52	19	EB	2A	53	37	10A
1B90	4F	33	C9	0F	28	43	23	CD	52	19	EB	2A	53	37	47	10A	105
Sum	CB	A1	B0	5E	08	44	63	18	B3	A0	5B	C1	7C	B9	4B	EB	18B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
1BA0	FA	50	18	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	17A
1BB0	E6	07	20	F5	CD	70	17	FE	08	2B	3A	FE	0D	20	33	FE	17A
1BC0	41	05	CD	99	19	ED	0F	1F	33	27	18	7C	B5	CA	59	13	17A
1BD0	33	E5	54	50	EB	CD	94	17	DA	59	13	ES	CD	7C	18	7C	17A
1BE0	66	19	C9	21	76	33	CD	ED	2A	CD	1A	33	CB	FE	27	26	17C
1BF0	CD	F6	17	21	77	33	0E	50	7E	FE	0D	20	33	FE	03	28	17C
1C00	23	18	F5	41	CD	99	19	ED	0F	1F	33	27	18	7C	B5	CA	187
1C10	16	C9	21	76	33	CD	ED	2A	CD	1A	33	CB	FE	27	26	45	187
1C20	72	00	18	23	CD	10	17	0D	53	7E	3A	74	61	78	20	45	19F
1C30	72	00	18	23	CD	10	17	0D	53	7E	3A	74	61	78	20	45	19F
1C40	72	00	18	23	CD	10	17	0D	53	7E	3A	74	61	78	20	45	19F
1C50	20	28	FA	FE	1B	28	FE	1B	28	FE	1B	28	FE	1B	28	FE	10A
1C60	7E	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	15F	15F	17D
1C70	30	7B	0F	FE	3A	38	FE	3A	38	FE	3A	38	FE	3A	38	FE	19A
1C80	37	C9	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	15F	19A
1C90	19	0D	7E	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	19A
Sum	3A	6A	21	B3	B1	76	EB	47	B1	B5	08	AB	E3	AA	0B	A3	1EC

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
1CA0	08	FE	3A	38	FE	3A	38	FE	3A	38	FE	3A	38	FE	3A	38	15A
1CB0	ED	7E	FE	08	2B	3A	FE	0D	20	33	FE	03	28	35	15F	15F	15A
1CC0	11	47	FE	01	21	FE	08	16	11	10	10	10	4B	00	E5	C1	105
1CD0	E5	1A	47	FE	01	21	FE	08	16	11	10	10	4B	00	E5	C1	105
1CE0	13	17	0F	1B	EB	78	18	7C	18	7C	18	7C	18	7C	18	7C	106
1CF0	20	0F	4E	1C	FE	29	82	FE	36	42	90	41	C9	E1	C9	189	
1D00	DA	59	13	FE	2C	28	05	CD	1C	3D	08	03	13	20	CD	B1	187
1D10	44	52	80	44	4A	5A	80	44	4A	5A	80	44	4A	5A	80	44	132
1D20	57	B1	44	4A	5A	80	44	4A	5A	80	44	4A	5A	80	44	4A	132
1D30	45	51	B1	44	4A	5A	80	44	4A	5A	80	44	4A	5A	80	44	132
1D40	4C	80	4F	55	54	C1	49	4E	50	48	40	52	54	54	80	A0	15D
1D50	52	54	80	4F	55	54	C1	49	4E	50	48	40	52	54	54	80	15D
1D60	80	41	44	43	80	53	42	9B	53	42	9B	53	42	9B	53	42	109
1D70	A0	58	4F	52	48	4F	52	48	4F	52	48	4F	52	48	4F	52	109
1D80	45	43	85	42	49	54	50	52	45	53	45	54	50	52	45	109	17F
1D90	4C	43	80	52	43	88	52	4C	90	52	52	58	93	43	41	108	108
Sum	5C	AB	10	AC	30	15	39	BA	B4	17	AD	0E	0A	ED	1F0	1F0	15D

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
1DA0	A0	53	52	41	AB	53	52	4C	BB	4C	44	49	43	50	49	12C	12C
1DB0	A1	49	4E	A2	4F	55	54	A3	4C	44	4A	AB	43	50	A16	16	16
1DC0	4A	A9	49	4E	4A	AA	4F	55	54	4A	AB	4C	44	49	52	B0	13A

1DD0	A3	50	49	52	B1	49	4E	49	52	B2	4F	54	49	52	B3	4C	100
1DE0	44	4A	52	B8	43	50	44	52	B9	49	4E	4A	52	BA	4F	54	100
1DF0	44	52	BB	4E	45	47	CA	52	4E	4E	C5	52	45	54	49	121	
1E00	CD	52	52	4A	47	52	4C	4A	4F	45	58	58	49	49	F3	166	
1E10	44	4A	52	BA	43	50	44	52	BA	49	4E	4A	52	BA	4F	166	
1E20	44	41	41	47	43	50	4C	4F	53	43	4B	52	43	43	4B	191	
1E30	48	41	4C	54	F6	52	4C	43	41	B7	52	47	43	41	8F	252	
1E40	4C	41	97	52	52	41	9F	41	80	42	81	43	82	43	83	45	
1E50	84	48	95	4C	86	28	4B	4C	28	28	28	28	28	28	28	28	
1E60	44	4A	52	BA	43	50	44	52	BA	49	4E	4A	52	BA	4F	197	
1E70	42	43	90	4A	45	41	4B	4C	92	53	50	93	49	58	94	109	
1E80	59	95	41	46	96	28	49	48	92	97	58	49	59	29	98	141	
1E90	46	27	99	28	53	50	29	9A	94	43	29	9B	4E	5A	5A	165	
Sum	E3	15	CB	98	28	28	BA	BD	31	23	01	43	CO	70	D2	94	

EDASマシン断ダンブ・リスト

```

2CFO 38 02 E6 7F DD 77 0B C3 55 28 A7 37 30 A7 C0 DD 190
2D00 7E 01 38 02 E6 7F C3 0A 28 3D C0 DD 56 01 CB BA 1C9
2D10 1E C0 DD 7E 03 FE 10 2B 16 1E DD FE 11 2B 10 1E 1D8
2D20 E0 FE 12 2B 0A CD 96 2D 2B 0A 1E F0 FE 16 C0 7B 141
2D30 B2 C3 0A 2B DD 77 0A 7B B2 1B 89 3D E6 07 07 07 13B
2D40 07 C9 DD 23 C0 5B 20 20 04 ED 53 CE 33 DD 2B C0 15C
2D50 5B 2D C0 ED 53 C0 33 C9 11 00 0A DD 7E 03 D6 17 1B3
2D60 2B 03 14 3D C0 DD 72 03 53 C9 D6 0A 2B 03 3D A7 199
2D70 C0 7E FE 2B 2B 0A FE 2B 2B 02 37 C9 C0 DA 1E BB 187
2D80 7E FE 29 37 C0 23 C3 E3 2A FE 0A 2B 06 FE 0B C0 18B
2D90 3E FD C2 3E DD C9 FE 14 2B F9 FE 15 1B F1 D5 F5 1FA
Sum 58 1B EA 9B 27 27 74 BB 13 E6 D2 67 56 C0 9C 4D 102

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
2DA0 81 4F F1 CD 66 32 7A CD B1 16 A7 CD BF 21 DA 0B 13D
2DB0 13 7B CD B1 16 A7 CD BF 21 DA 0B 13 D1 C9 0E 01 E6
2DC0 3E 3A A7 CD BF 21 DA 0B 13 D1 65 33 66 7B CD 0E 176
2DD0 2D ED 5B 63 33 7A CD 9E 2D 7B CD 9E 2D AF CD 9E 14A
2DE0 2D 23 7E CD 9E 2D 10 F9 79 ED 4A CD 9E 2D CD 0F 18D
2DF0 17 2A 63 33 3A 65 33 5F 16 00 19 22 63 33 8F 32 1B0
2E00 65 33 21 66 33 22 61 33 C9 D9 F5 3A 65 33 A7 2B 140
2E10 0A 47 3A 65 33 80 E6 0F CE 0B 2D 2A 61 33 F1 77 173
2E20 23 22 61 33 21 65 33 3A D9 C9 3A 3B 33 CD B1 16 1A4
2E30 CD 0C 1C CD F6 17 2A 57 33 CD 7E 2D 0C 1C 06 1EB
2E40 0A DD 7E 09 A7 2B 0E 4F DD 7E 0A DD 23 CD 6A 2E 15E
2E50 05 05 0D 20 F3 CD 0C 1C 10 FB 2A 41 33 CD E7 1A 196
2E60 CD 7B 2E C9 D5 CD 66 32 7A CD B1 16 7B CD B1 16 196
2E70 D1 C9 7C CD 64 2E 7D CD 6A 2E C9 3A 33 FE 04 1C5
2E80 1B 5B E5 3A 3C 33 FE 3F 2B 4E 47 CD 0F 17 10 FB 1FE
2E90 3E 3F 32 3C 33 CD 15 17 20 20 50 41 47 45 20 194
Sum AD A3 C5 AC D5 14 E5 EA 55 BB 54 E9 FD A6 BD A0 163

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
2EA0 2A 3F 33 23 22 3F 33 CD F9 17 CD 15 17 0D 45 20 19B
2EB0 53 54 4E 52 20 41 44 52 53 20 4F 42 4A 45 43 54 16B
2EC0 20 20 20 20 53 4F 55 52 43 45 20 53 54 41 54 45 172
2ED0 4D 45 4E 54 53 0D 0B 01 C9 3D 32 3C 33 CD 0F 105
2EE0 17 C9 2A 41 33 23 7E FE FA 3B FA C9 CD B5 17 FE 179
2EF0 0B CA 0B 13 32 3A 33 FE 57 2B 0D FE 5A 2B 09 FE 19D
2F00 53 2B 05 FE 4C C2 59 13 CD 0D 1B CD 15 17 0D 45 1F8
2F10 5B 43 2F 4C 61 62 65 6C 20 54 61 62 6C 65 20 47 19
2F20 4D 00 AF 32 39 33 32 3B 33 01 00 0D 52 19 22 41 33 13C
2F30 ED 43 59 33 ED 5B 4F 33 13 0D CD 52 19 22 41 33 13C
2F40 CD F3 2A 2B 2B 2A 41 33 7E FE FF 2B 16 E1 23 ED 18C
2F50 5B 53 33 CD 94 17 D2 37 1C ED 5B 59 33 2B 73 23 113
2F60 72 23 E5 CD 66 19 30 17 CD E2 2E 1B 0D CD 15 17 1CB
2F70 0D 45 72 72 20 69 6E 20 00 CD F6 17 C3 47 1C E1 12E
2F80 22 5F 7A 3A 3A 33 FE 54 C2 57 30 CD 15 17 0D 45 141
2F90 5B 43 2F 4D 75 6C 74 20 4C 61 62 65 6C 20 43 4B 11A
Sum 15 B9 70 B7 11 4D EC 6C 69 F1 D6 06 F6 AB A5 4E 143

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
2FA0 00 2A 43 33 22 49 33 CD 52 19 D9 ED 5B 45 33 2A 139

```

```

2FB0 49 33 D9 2B 3E F9 23 BE D2 B6 2F D9 E5 A7 ED 52 1F3
2FC0 E1 23 D9 D2 3E 31 7E 3C CA B4 2F D9 E5 D9 C1 AF 18C
2FD0 32 37 33 3E 20 23 BE 30 FC E5 3E F9 23 BE D2 DC 1B2
2FE0 2F D9 E5 A7 ED 52 E1 23 D9 DA EF 2F C2 0E 30 7E 126
2FF0 3E CA DA 2F 3E 20 23 BE 30 FC D1 D5 1A FE 3A 2B 19A
3000 15 FE 21 3B 11 BE C2 DA 2F 23 13 C3 FC 2F E1 C5 10D
3010 D9 E1 D9 C3 E5 2F 7E FE 21 3B 05 FE 3A C2 DA 2F 115
3020 D9 22 49 33 D9 E5 3A 37 C3 A7 20 11 CD 15 17 0D 1B7
3030 2A 2A 2A 2A 2A 0D 00 69 6D 2B CD F9 17 CD 0F 17 1A3
3040 2A 49 33 2B CD F9 17 3E 0F 32 37 33 E1 1B 0B E1 1FC
3050 ED 43 49 33 C3 B3 2F CD 0F 17 21 00 00 22 57 33 111
3060 22 3F 32 22 55 33 3E 01 32 3C 33 3A 3A 33 32 39 130
3070 33 FE 57 CA 30 31 FE 4C CC B2 2E 2A 43 33 22 49 1B4
3080 33 CD 52 19 3E 20 32 3B 33 CD F3 2A DA 6D 2F CD 190
3090 66 19 30 05 CD E2 2E 1B EB 3A 39 33 F5 FE 57 20 1A4
Sum BD 34 DC 04 D0 F9 F2 FB 10 79 1F 55 6B 6D BA 4B 15E

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
30A0 10 3A 65 33 A7 C4 BE 2D 3E 3A A7 CD BF 21 DA 0B 1C6
30B0 13 0A 00 3E 30 A7 CD BF 21 DA 0B 13 10 F5 CD 15 1BA
30C0 17 0D 45 52 52 4F 52 53 20 00 2A 55 33 CD F9 17 1B0
30D0 CD 0F 17 F1 FE 53 C2 0B 13 3A 5B 33 A7 CA 0B 13 149
30E0 CD 15 17 0D 52 65 6C 6F 63 61 74 69 6F 6E 20 42 17B
30F0 69 61 73 2D 00 CD ED 5B 5B 33 A7 ED 52 23 4D 44 E1 1CF
3100 DF E5 2A 5D 33 ED 5B 5B 33 A7 ED 52 23 4D 44 E1 1CF
3110 E5 19 EB 2A 5F 33 CD 94 17 D2 2B 31 09 2B E3 ED 1FA
3120 5B 5D 33 19 EB E1 ED BB C3 B1 00 ED B0 C3 B1 00 14F
3130 CD F8 2D AF 37 CD BF 21 DA 0B 13 C3 7B 30 3E 5A 150
3140 32 39 33 2A 45 33 23 22 45 33 CD 0F 17 CD 0F 17 1E3
3150 3A 3D 33 47 32 3E 33 3E 3F 32 3C 33 CD 15 17 53 1FE
3160 59 4D 42 4F 4C 20 41 44 52 53 20 20 00 10 ED CD 1D7
3170 0F 17 CD 0F 17 2A 43 33 22 49 33 CD 52 19 01 FF 18F
3180 FF 2B 19 32 D2 13 32 E5 DD 0D E1 CD 4D 1C C5 E5 1D7
3190 CD 19 32 30 25 E5 FF E1 CD 4D 1C D1 D5 1E BE 3B 11C
Sum D6 46 34 55 E5 7F 4A 5D 84 1A D6 DC AA C5 05 21 1BE

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
31A0 EF 20 10 FE 1B 2B E9 FE 20 2B E5 FE 3A 2B E1 23 1D8
31B0 13 1B EA F1 F1 FD E5 E1 1B ED DD 36 00 FA 06 06 1B9
31C0 E1 7E FE 1B 2B 0E FE 2B 0A 0A FE 3A 2B 06 CD B1 1E2
31D0 16 23 10 ED 0A CD 0C 1C 10 FB C1 2A 4F 33 23 09 1D3
31E0 09 5E 23 5A EB CD 72 2E CD 0C 1C CD 0C 1C 3A 3E 19A
31F0 33 3D 32 3E 33 C2 F5 31 3A 3D 33 32 3E 3C 3A 3E 13C
3200 33 3D 32 3E 33 FE 03 C2 72 31 47 CD 0F 17 10 FB 13B
3210 C3 4A 31 CD 0F 17 C3 0B 13 CD BB 1C DD 03 7E FE 1D2
3220 FA 2B F6 37 C9 D5 E5 F5 3A 3B 33 A7 2B 2B EB 2A 17B
3230 5D 33 ED 52 D2 37 1C ED 22 5D 33 ED 5B 5B 33 A7 10E
3240 ED 52 ED 5B 5F 33 19 DA 37 1C ED 5B 53 33 CD 94 1BE
3250 17 D2 37 1C F1 77 E1 D1 C9 22 5B 33 22 5D 33 3E 1BF
3260 0F 32 3B 33 1B D5 F5 E6 0F CD 79 32 5F F1 E6 F0 121
3270 0F 0F 0F CD 79 32 57 C9 FE 0A 30 03 F6 30 C9 1FE
3280 C6 37 C9 00 00 00 00 00 00 00 00 00 00 00 00 00 1C6
Sum 6A F2 D7 D6 6B AB A7 0F 30 E0 D0 04 34 C1 0D B2 167

```

MZ-80B



ラベル付き

逆アセンブラ

COMPAC

表1 ワーク・エリア対照表

ワーク・エリア名	K/C	目	内 容
BUFER	11A3	1093	キー入力バッファ80バイト
IBUFE	10F0	1140	ファイル・モード
NAME	10F1	1141	ファイル・ネーム
SIZE	1102	1142	ファイルのバイト数
DTADR	1104	1154	データ開始アドレス
EXADR	1106	1156	スタート・アドレス

今回はMZ-80K/Cの“ラベル付き逆アセンブラ”(81年2月号)をMZ-80Bに移植したので、その移植のテクニックも含めて報告します。

移植テクニック

K/Cタイプのマシン語のロード方法については、“テープ・フォーマットの変換法”で大畑氏が調べているので参照してください。

ロードした後はモニタのMコマンドでK/Cタイプのサブルーチン、ワーク・エリアのアドレスをBタイプのアドレスに変換すればよいわけです。

ほとんどのサブルーチンはK/Cタイプと同じ仕様になっていますが、いくつか違うものもあります。たとえばカセットへのリード、ライト・ルーチンはレジスタが保存されないでコールする前にPUSHしておく必要があります。

また、ディスプレイ・コード←ASCII変換はBタイプでは不用となります。BタイプではキャラクタはすべてASCII形式で表わされています。

ゲームなどのキャラクタはコード表を見ながら書き換えてください。V-RAMを直接アクセスする場合には、PIOのポートE 8の最上位ビットを立ててV-RAMに切り替える必要があります。

これはBASICの場合と同じですが、LIMIT文を実行すれば切り替えた後にPRINT文を実行すると自動的に元の状態に戻ってくれます。しかし、マシン語の場合はアクセスした後にまた元に戻してやらないと暴走する可能性があります。プリンタ出力ルーチンに関してはほとんど変更なく使えますが、K/Cタイプの場合キャリッジ・リターン・コード(0D)が送られると改行も行なわれますが、Bタイプの場合はライン・フィード・コード(0A)が送られないと改行しないようになっています。したがって、0DHが送られてきたら0Aに代えてやらなくてはなりません。

アドレスをバッチするだけでもある程度は動作します。K/CタイプにあってBタイプにないサブルーチンは自分で作るしかありません。

K/Cタイプが使える場合には、シャープから供給されているシステム・プログラム(エディタ・アセンブラなど)を使ってさらに効率よく移植できます。

BタイプではK/Cタイプのテープも作れるので、このラベル付き逆アセンブラを使ってK/C用のソース・テープを作り、K/Cのエディタの修正コマンドを使ってモニタ・サブルーチンのアドレスを変えてからアセンブルすることによってBタイプ用のオブジェクト・プログラムが作れるわけです。

Bタイプのテープ入力をK/Cモードに切り替えておけばK/Cで作成したオブジェクト・テープを走らせることができるわけです。もっとも、近くBタイプ用のシステム・プログラムがシャープから供給されるはずなので、それを使えばすべてBタイプだけで効率よく移植できます。

MZ-80Bのモニタのサブルーチンの仕様がほとんどK/Cと同じになっているということは、いままではK/Cで開発されたソフトがムダにならないので非常に助かります。

また、表1にワーク・エリア対照表を載せておきます。

使用方法

スタート・アドレスは\$8C00です。起動させると“MODE”と表示し入力待ちになります。MODEは表3のよう

MODE 1

ラベルなしの単なる逆アセンブラでスタート・アドレス(アセンブル開始アドレス)をキーインするとリストを出力します。

MODE 2

ラベルなしの逆アセンブラですが、プログラムを本来格納すべきアドレスと異なるエリアに格納して逆アセンブル・リストを出力します。つまり転送されたプログラムも元のアドレスからの逆アセンブル・リストとして出力します。LOADアドレスとは実際にマシン語が格納されているア

表2 プログラムの配置

アドレス	内 容	アドレス	内 容
8C00~8C56	MODE入力処理	9348~9376	ADD HL, ss : SBC HL, ss ADC HL, ss : LD (nn), ss ADD A, n-CP n その他のモニツク・コード
8C57~8CE5	MODE1処理	9377~93A5	オペランドなし
8CE6~8CEE	MODE2処理	93A6~949A	CB系コード
8CEF~8D89	MODE3処理	949B~948B	DD, FD系コード
8DA~8E72	MODE4処理	948C~9505	ED系コード
8E73~8EBE	テープ出力用コード作成転送	9506~954B	オペランドn, nn, eなどの処理
8EE0~902F	MODE T処理	954C~955B	コード転送サブルーチン
9031~9037	MODE C処理	954C~962F	テーブル検索, アドレス入力など
9038~9045	MODE #処理	9530~96AD	バイト数処理テーブル (3つ)
9046~9077	16進コードによる命令ごとのダンブ	9734~978A	ラベル必要コード (2つ)
9078~910D	命令バイト数判断	978B~97A9	モニツク・コード処理先テーブル
910E~91EF	ラベル処理	97AA~98A1	オペランドなしコード・テーブル
91F0~9228	ニモニツク・コード作成メイン	98A2~98C4	オペランド検索テーブル
9229~924F	DEFBの処理	98C5~993B	メッセージ・エリア
9250~926A	LD r, r'	993D~9913	モニツク・コード・テーブル
926B~9295	ADD A, r: OUT(c) r INC r: DEC r	9A14~9C18	ワーク・エリア
9297~9306	RET cc: RST nn: LD r, n: JP cc, nn: CALL cc, nn: IN r, (c)	9C19~9CFE	プリンタ・コントロール
9307~9347	INC ss: DEC ss: POP qq. PUSH qq: LD dd, nn.	9D00~9F03	メッセージ, テーブル, ワーク・エリア
		9F04~9F5E	スタック・エリア
		9F5F~9FFF	

表3 逆アセンブラのモード

MODE	機 能	キー・イン事項と順序	出力対象
1	ラベルなしの通常の逆アセンブル	(1) START ADDR.	テレビ プリンタ
2	ラベルなし, 逆アセンブルされるプログラムを本来の格納アドレスと異なるエリアに格納しても逆アセンブルできる。	(1) LOAD ADDR. (2) REAL ADDR. (3) START ADDR.	テレビ プリンタ
3	ラベル付きで, MODE 2 タイプの逆アセンブラ。 (LABEL TABLE ADDR. は2000H以上に設定のこと)	(1) LOAD ADDR. (2) REAL ADDR. (3) LABEL TABLE ADDR. (4) LABEL START ADDR. (5) LABEL END ADDR. (6) START ADDR.	テレビ プリンタ
4	ラベル付きで, MODE 3タイプの逆アセンブラ。ただし, 出力はカセット・テープだけである。	(1) LOAD ADDR. (2) REAL ADDR. (3) LABEL TABLE ADDR. (4) LABEL START ADDR. (5) LABEL END ADDR. (6) CODE TOP ADDR. (7) START ADDR. (8) END ADDR. (9) FILENAME.	カセット・ テープ
T	定数エリアの登録, 抹消および定数エリア・テーブルの表示。	NN_MM_XXXX_YYYY 01 1 2000 2080 CRで登録。 ↑ ↑ 登録エリア 登録番号 登録(1) 抹消(0) 空白エリアにCRでMODEキー・イン待ち	
C	定数エリア・テーブルのクリア (全解除)		
#	逆アセンブル・リストをプリンタに出力するか, 停止するかを決定, #をキー・インすることにより状態反転。		
M	MODE選択キー入力待ちへジャンプする。	注) 9771A以下2バイトの内容を変えると, その内容によって決まるアドレスへジャンプ先が変更される。	

備考 LOAD ADDR. →逆アセンブルされるプログラムの先頭アドレス
 REAL ADDR. →本来格納されるべきアドレスの先頭アドレス
 LABEL TABLE ADDR. →ラベル・テーブルの先頭アドレス
 LABEL START ADDR. →ラベルを付ける範囲の始りのアドレス
 LABEL END ADDR. →ラベルを付ける範囲の終わりのアドレス
 CODE TOP ADDR. →逆アセンブル・リストをASCIIコードで格納し, テープへSAVEするために一時格納するためのエリアの先頭アドレス
 START ADDR. →逆アセンブル開始アドレス
 END ADDR. →逆アセンブル終了アドレス

ドレスで、REALアドレスとは本来格納すべきアドレスのことです。

■MODE 3

ラベルの付いた逆アセンブル・リストを出力します。LO ADアドレスとREALアドレスはMODE 2と同じです。転送されていない場合には同じ値をキーインしてください。

ラベル・テーブルを作るのでじやまにならないところにラベル・テーブル・アドレスを指定してください。次にラベルを付ける範囲をキーインします。

■MODE 4

MODE 3と同じくラベル付き逆アセンブラですが、出力はカセットだけです。RAM上にテキスト・エディタと同じフォーマットのASCII列で落とし、それをカセットにセーブする方式をとっています。内部メモリの規模を考えてコード・トップ・アドレスを決めてください。

■MODE T

データ・エリアがわかっている場合、そのエリアを登録できます。その場合、逆アセンブル時の出力は例1のようになります。

プリンタとの接続

プリンタへの出力または停止は、MODE選択時、もしくは「スペース」キーを押して出力リスト停止状態にあるときに「#」をキーインすると、出力または停止の状態を反転させることができます。プリンタ出力状態ではリスト50行で自動的にフォーム・フィードするようにしてあります。

本プログラムの971A番地以下、2バイトの内容を変えると「SHIFT BREAK」を押したとき（ただし、リスト出力が停止している状態）でのジャンプ先アドレス（通常、MODEキー入力待ちに設定されている）が変更できます。

なお、プリンタ・コントロールの機能を表5に示します。

□参考文献

北村元二：“ラベル付き逆アセンブラ”，I/O，'81年2月号

例1 モード3実行時

000D 00	DEFB 00H
000E 00	DEFB 00H
000F 00	DEFB 00H
0010 C3B100	JP \$00B1
0013 00	DEFB 00H
0014 D0	DEFB D0H
0015 FF	*0015:DEFB FFH
0016 00	DEFB 00H
0017 00	DEFB 00H
0018 C3B100	JP \$00B1
001B C0	DEFB C0H
001C 03	DEFB 03H
001D 04	*001D:DEFB 04H
001E 76	DEFB 76H
001F FF	DEFB FFH
0020 C3B100	JP \$00B1

表4 リスト出力一時停止のときの操作キー

操作キー	機	能
スペース		リスト出力の再開、停止
SHIFT/BREAK		逆アセンブル中止、MODE選択キー入力待ちにジャンプ
N		逆アセンブル・アドレスの変更
P		プリンタ・コントロール・ルーチン呼び出し
R		MODE選択キー入力待ちにジャンプ
#		プリンタ出力または停止状態を反転

(*) 971A以下2バイトのアドレス内容を変え、ジャンプ先が変更される

表5 プリンタ・コントロールの操作キー

操作キー	機	能
シフト+H		倍文字表示（解除）
シフト+M		132モード（解除）
H		ホーム
T		全モード解除
N		48行スキップ
L		改行（して繰り返し）（シフト・ブレークでRET）
N		改頁
M		メッセージ・プリント（Mで繰り返し）（シフト・ブレークでRET）



ラベル付き逆アセンブラ メモリ・ダンプ

BC00 71 00 A0 CD 2E 0A 11 7D :24	BCD8 D4 3A 7C 9C EE 01 32 3C :43	B0B0 89 08 CD CD 96 ED 58 1E :27
BC08 99 CD 89 08 11 19 9C 06 :C7	BCE9 9C CD 14 0F 1B C7 CD BF :F7	B1B8 9C AF ED 52 22 29 9C CD :3E
BC10 4B AF 12 17 10 FC 31 00 :5C	BCEB 8E 21 00 00 C3 5D 8C AF :0A	BDC0 2E 0A 11 9F 99 CD 89 08 :0F
BC18 A0 CD 2E 0A 11 76 99 CD :92	BCF0 32 2F 9C CD BF 8E CD 2E :12	BDD8 CD CD 96 ED 58 1E 9C AF :E1
BC20 89 08 CD CD 08 12 97 A0 :9C	BCFB 0A 11 78 99 CD 89 08 CD :5A	BDD0 ED 52 22 7A 9C CD 2E 0A :3C
BC28 F5 CD C6 08 F1 FE 31 CA :7A	BDD0 CD 96 22 9C 00 11 00 20 :74	BDD8 11 E7 99 CD 89 08 2A 29 :A2
BC30 57 8C FE 32 CA E6 8C FE :4D	BDD8 CD EA 91 38 E9 AF 77 25 :B2	BDE0 9C ED 58 7A 9C 00 EA 91 :02
BC38 71 0A EF 8C FE 34 CA 8A :FE	BDD8 77 32 28 9C 21 00 00 22 :B0	BDE8 50 9E CD 46 90 CD 0E 91 :4D
BC40 BD FE 54 CA E0 8E FE 47 :58	BDD8 24 9C 22 26 9C CD 2E 0A :A9	BDF0 CD F0 91 CD 77 8E 18 E6 :1A
BC48 7A 70 90 FE 4D CA 19 97 :4F	BDD8 11 8D 99 CD 89 08 CD :2F	BDF8 CD 14 0F CD 2E 0A 11 FD :03
BC50 5F 27 CA 18 90 1B 8F 21 :A8	BDD8 96 22 24 9C ED 58 1E 9C :7A	BEO0 99 CD 89 08 CD 2E 0A 11 :0D
BC58 00 00 22 1E 9C 22 24 9C :BE	BDD8 AF ED 52 22 29 9C CD 2E :0D	BEO8 93 10 CD A4 06 11 40 11 :7C
BC60 22 26 9C AF 32 28 9C 7E :C7	BDD8 0A 11 8C 99 CD 89 08 CD :B8	BE10 7E 03 12 17 21 93 10 01 :28
BC68 01 72 21 9C CD 2E 0A 11 :06	BDD8 00 26 22 26 9C E5 ED 58 :74	BE18 10 00 ED 80 3E 0D 12 2A :34
BC70 9C 99 CD 89 08 CD 96 89 :79	BDD8 1E 9C AF ED 52 22 28 9C :91	BE20 7A 9C 22 54 11 22 56 11 :E0
BC78 E5 7A 3C 9C FE 01 CD C3 :85	BDD8 ED 58 24 9C E1 CD EA 91 :31	BE28 E8 2A 76 9C 36 FF 27 AF :EE
BC80 9D E1 ED 58 1E 9C AF ED :1C	BDD8 78 8A CD 2E 0A 11 BE 99 :5F	BE30 ED 52 22 78 9C CD 51 02 :55
BC88 52 22 29 9C CD 46 90 CD :A9	BDD8 CD 89 08 2A 29 9C ED 58 :95	BE38 78 26 2A 54 11 11 00 01 :FF
BC90 0E 91 CD F0 91 2E 0A FE :B2	BDD8 2B 9C CD EA 91 30 08 CD :1A	BE40 AF ED 52 22 54 11 2A 54 :F7
BC98 11 70 9C CD 89 08 7A 3C :F2	BDD8 78 9C CD 2E 91 18 EC 3E :06	BE48 11 19 22 54 11 2A 38 9C :AF
BCA0 9C FE 01 CD D4 9D CD 32 :07	BDD8 01 32 28 9C CD 14 0F :A2	BE50 AF ED 52 FA 65 8E 22 38 :35
BCA8 08 FE 20 2D CD 32 08 :2C	BDD8 2F 9C FE 0A CA 92 BD C3 :79	BE58 9C ED 57 52 11 CD E4 9E :BE
BCB0 FE 00 20 F9 CD 62 05 CA :15	BDD8 67 8C 3E 0A 32 2F 9C :F5	BE68 DA 28 97 18 E1 00 00 00 :92
BCB8 19 97 CD 32 08 FE 20 28 :FD	BDD8 F7 8C CD 2E 0A 11 05 99 :03	BE68 7A 00 90 CD 82 02 3A 0A :7A
BCC0 CB FE 4E 2B A2 FE 52 CA :F8	BDD8 89 08 CD 82 26 22 3A :E4	BE70 C1 1C 97 ED 58 76 9C 21 :81
BCC8 16 8C FE 50 28 0E FE 23 :3F	BDD8 9C 22 76 9C 3E 01 72 21 :22	BE78 AB 9C 7E 7E 2A 28 05 01 :B1
BCD0 2B 07 18 E0 CD 42 9D 1B :E8	BDD8 9C CD 2E 0A 11 92 99 CD :AA	BE80 06 00 ED B0 21 31 9C 7E :2F

8E88	FE	20	28	04	ED	A0	18	F7	E6	9158	B5	28	15	CD	D5	91	58	0E	:69	93E8	9B	D5	CD	33	96	E1	CD	3D	:F1	
8E90	21	56	9C	7E	FE	20	28	1E	:F5	9140	CD	53	91	28	09	EB	73	23	:163	93F0	96	C9	CD	C1	95	21	72	9B	:B0	
8E98	7E	20	12	13	7E	FE	20	28	:47	9148	72	25	AF	77	27	7D	DD	:E1	:13	93F8	11	D8	9B	18	EC	CD	C1	95	:AB	
8EA0	04	ED	A0	18	F7	21	5D	9C	:BA	9150	E1	01	C9	DD	2A	22	9C	DD	:1D	9400	21	6F	9B	18	F3	CD	81	95	:19	
8EAB	7E	FE	20	09	7E	FE	20	17C		9158	5E	00	DD	56	01	78	82	:28	:E7	9408	21	58	9B	11	CA	9B	18	D9	:7E	
8EB0	28	04	ED	A0	18	F7	3E	00	:13	9160	08	CD	23	18	EB	3C	DD	D5	:D2	9410	CD	81	95	21	5E	9B	18	F3	:08	
8EB8	12	1E	ED	53	36	9C	C9	CD	:CD	9168	DD	23	18	EB	3C	DD	D5	D1	:D2	9418	11	34	9B	21	77	9B	18	C9	:F4	
8EC0	2E	0A	11	52	99	CD	89	08	:1E	9170	C9	D5	CD	A3	91	3A	1D	9C	:92	9420	11	36	9B	18	F6	11	38	9B	:04	
8EC8	CD	96	E5	CD	2E	0A	11	:28		9178	FE	00	28	21	FE	02	20	05	:6C	9428	18	F1	21	7A	9B	11	F7	9A	:E1	
8ED0	64	99	CD	89	08	CD	96	:88		9180	CD	17	96	18	15	FE	03	20	:C8	9430	01	A0	9B	E5	D5	CD	33	96	:79	
8ED8	D1	AF	ED	52	22	1E	9C	C9	:64	9188	06	ED	5B	19	9C	18	05	ED	:0D	9438	E1	CD	30	96	E1	CD	4B	96	:10	
8EE0	CD	2E	0A	11	F9	9B	CD	89	:00	9190	58	19	9C	13	13	EB	5E	23	:A2	9440	C9	CD	75	95	21	47	9B	11	:BA	
8EE8	08	7A	65	9C	FE	00	28	9C	:C5	9198	56	EB	AF	01	C9	21	00	00	:AB	9448	CA	9B	01	BA	9A	18	E4	CD	:7D	
8EF0	47	0E	01	DD	21	66	9C	11	:67	91A0	37	18	F8	ED	E5	3A	1C	9C	:F3	9450	75	95	21	63	9B	11	BA	9A	:BE	
8EF8	91	10	79	CD	90	96	21	EA	:27	91AB	FE	01	28	18	ED	5B	19	9C	:F3	9458	01	C3	96	18	D6	11	9F	9B	:99	
8F00	98	CD	76	96	DD	E0	00	DD	:9C	91B8	1A	FE	DD	28	19	FE	ED	28	:49	9460	21	7A	9B	01	DE	9B	18	CB	:93	
8F08	66	01	CD	94	96	3E	20	12	:CE	91BB	15	FE	DD	28	11	21	BB	97	:8C	9468	11	F1	9A	18	F3	21	44	9B	:A7	
8F10	17	DD	6E	02	DD	66	05	CD	:73	91CD	CD	EF	90	E1	01	20	01	AF	:18	9470	11	F4	9A	01	DE	9B	18	BB	:EC	
8F18	14	96	3E	00	12	CD	2E	0A	:8C	91CB	32	1D	9C	E1	D1	C9	13	1A	:93	9478	11	7D	9B	01	BA	9A	21	44	:E3	
8F20	11	93	10	CD	89	08	CD	32	:11	91DD	21	9E	97	18	EB	05	ED	5B	:76	9480	9B	18	B0	11	BA	9A	01	7D	:46	
8F28	08	7E	20	20	14	CD	32	08	:61	91DB	24	9C	CD	EA	91	38	09	ED	:56	9488	9B	18	F3	11	7F	9B	01	BA	:8C	
8F30	7E	20	28	0D	CD	72	08	FE	:58	91ED	58	26	9C	EB	CD	EA	91	EB	:78	9490	9A	18	EB	11	BA	9A	01	7F	:82	
8F38	20	28	06	FE	08	28	DD	18	:AA	91EB	D1	C9	7C	BA	CD	7D	BB	C9	:91	9498	9B	18	E3	3A	20	9C	21	A2	:4F	
8F40	ED	0C	DD	23	DD	23	DD	23	:FF	91F0	3A	1C	9C	FE	01	20	04	CD	:E2	94A0	9B	06	01	BE	28	04	04	23	:80	
8F48	DD	23	10	AB	CD	2E	0A	11	:D1	91FB	29	92	C9	3A	21	9C	FE	01	:7A	94AB	18	F9	21	14	9A	7E	FE	DD	:69	
8F50	93	10	CD	44	06	3A	96	10	:FA	9200	20	16	21	8D	9B	11	DE	9B	:09	94B0	28	03	23	18	F8	10	F8	23	:8C	
8F58	FE	20	28	54	FE	31	2B	0C	:0D	9208	CD	8C	96	21	A4	9B	11	E1	:41	94BB	CD	33	96	C9	ED	5B	19	9C	:5C	
8F60	FE	20	CA	16	BC	FE	00	CA	:5F	9210	9B	CD	8C	96	9F	32	21	9C	:28	94C0	3A	18	9C	FE	04	20	9C	21	:4E	
8F68	16	BC	18	E3	11	98	10	CD	:23	9218	ED	5B	19	9C	1A	32	20	9C	:05	94CB	13	13	1A	32	20	9C	21	4E	:9D	
8F70	14	06	DA	A7	BF	E5	11	9D	:80	9220	F5	21	AA	97	CD	AE	96	F1	:59	94D0	9B	CD	AE	96	E9	CD	57	96	:4C	
8F78	10	CD	14	06	D1	DA	A7	BF	:D8	9228	E9	10	90	9B	CD	33	96	ED	:B8	94D8	21	1D	99	CD	30	96	3A	20	:C4	
8F80	CD	EA	91	DA	A7	BF	7A	65	:F7	9230	5B	19	9C	1A	F5	11	56	9C	:22	94E0	9C	E6	07	21	C5	9B	CD	3A	:0E	
8F88	9C	07	07	AF	06	00	DD	21	:DF	9238	CD	9D	96	3E	4B	12	F1	FE	:87	94EB	96	C9	21	81	9B	CD	33	96	:32	
8F90	66	9C	DD	09	DD	73	00	DD	:15	9240	20	DD	08	FE	60	DD	11	5D	9C	:30	94F0	CD	57	96	21	2D	9B	CD	3A	:AB
8F98	72	01	DD	75	02	DD	74	03	:1B	9248	F5	2E	3B	12	13	F1	12	C9	:9F	94FB	96	C5	5F	92	21	85	9B	18	:A3	
8FA0	21	65	9C	3A	C3	4C	BF	11	:05	9250	21	44	9B	CD	33	96	CD	57	:BA	9500	EC	21	89	9B	ED	17	3E	01	:6F	
8FAB	EE	98	CD	89	08	CD	96	BF	:85	9258	96	21	C5	9B	CD	3A	96	3A	:EB	9508	32	21	9C	3A	20	9C	FE	DD	:C0	
8FAC	11	93	10	CD	23	06	78	EF	:01	9260	20	9C	E6	07	21	C5	9B	CD	:F4	9510	20	04	3E	3A	28	18	02	3E	:59	
8FAD	47	FE	00	CA	A7	BF	7A	65	:E4	9268	48	96	C9	CD	57	96	21	DD	:8F	9518	32	85	9B	32	B9	9B	CD	0B	:E0	
8FAE	9C	90	9A	DA	A7	BF	CA	00	:F5	9270	99	CD	30	96	3A	20	9C	4F	:71	9520	96	21	84	9B	11	DE	9B	CD	:5D	
8FBC	07	07	AF	78	3D	06	00	07	:1F	9278	EE	80	E6	F0	28	11	79	FE	:E4	9528	8C	96	21	87	9B	11	E1	9B	:22	
8FBD	07	5F	16	00	21	66	9C	19	:8B	9280	9B	E6	F8	28	0A	79	E6	07	:0E	9530	CD	8C	96	ED	5B	19	9C	13	:FF	
8FBE	E5	23	23	23	D1	ED	80	DD	:DF	9288	21	C5	9B	CD	3A	96	C9	21	:05	9538	1A	F5	FE	E9	20	0A	3E	29	:87	
8FB0	21	65	9C	75	C3	4C	BF	11	:05	9290	BA	9A	CD	3D	96	18	CB	21	:F5	9540	32	E4	9B	3E	28	32	E5	9B	:AE	
8FB8	EB	3A	65	9C	FE	00	28	28	:74	9298	47	9B	CD	33	96	21	A9	9B	:8D	9548	F1	C3	1D	92	ED	5B	19	9C	:60	
8FBC	47	DD	21	66	9C	DD	5E	00	:82	92A0	CD	30	96	CD	57	96	C3	64	:81	9550	13	1A	32	20	9C	21	5E	9B	:32	
8FBD	00	5A	01	ED	53	30	9C	DD	:1D	92AB	92	21	4B	9B	11	C5	9B	01	:08	9558	CD	AE	96	E9	DD	2A	19	9C	:B6	
8FBE	5E	92	DD	56	05	CD	53	32	:08	92B0	81	9B	C5	D5	CD	33	96	CD	:49	9560	3A	18	9C	FE	04	20	04	DD	:F4	
8FBF	9C	DD	1B	90	30	DD	DD	23	:4F	92B8	57	96	E1	CD	3A	96	E1	CD	:19	9568	23	DD	23	DD	7E	01	11	CD	:50	
8FC0	DD	23	DD	23	DD	23	DD	23	:ED	92C0	48	96	C9	21	4F	9B	18	EA	:81	9570	9B	CD	90	96	C9	DD	2A	19	:8A	
8FC1	7D	C9	D5	ED	5B	30	9C	9A	:BA	92CB	21	53	9B	11	D5	9B	18	DF	:84	9578	9C	DD	7E	01	11	C5	9B	18	:B1	
8FC2	CD	EA	91	78	09	ED	5B	32	:03	92D0	21	57	9B	11	E5	9B	18	D7	:90	9580	F0	DD	E5	D5	E5	11	CA	9B	:E2	
8FC3	0E	ED	CD	EA	91	EB	D1	C9	:54	92DB	CD	5C	95	21	44	9B	11	C5	:94	9588	DD	2A	19	9C	3A	18	9C	FE	:AB	
8FC4	AF	21	65	9C	E5	16	BC	9C	:08	92E0	9B	01	C0	9B	18	CC	DD	B1	:26	9590	04	20	02	DD	23	DD	6E	01	:72	
8FC5	7A	7C	9C	EE	01	32	3C	9C	:08	92E8	95	21	5B	9B	11	D5	9B	01	:2B	9598	DD	66	02	CD	D5	91	38	10	:C0	
8FC6	CD	14	0F	C3	16	BC	DD	69	:88	92F0	CA	9B	18	BE	CD	81	95	21	:F3	95A0	7C	85	28	0C	3E	24	12	13	:EC	
8FC7	96	CD	78	90	2A	19	9C	E5	:29	92F8	5E	9B	18	F0	21	63	9B	11	:31	95AB	CD	94	96	E1	D1	DD	E1	C9	:30	
8FC8	ED	5B	1E	9C	19	11	30	9C	:05	9300	C5	9B	01	A9	9B	18	AB	21	:86	95AC	DD	94	96	3E	48	12				

9698 7D C8 9D 96 C9 F5 0F 0F :59	9948 41 53 53 45 44 42 4C 45 :4C	97BF 0D 4E 4E 4E 20 4D 20 58 58 :E6
96A0 0F 0F C8 F3 05 12 17 F1 :F9	9950 52 0D 4C 4F 41 40 20 :BF	9C00 58 58 20 59 59 59 59 :41
96AB C8 F3 05 12 13 C9 C5 D5 :AD	9958 20 20 20 20 20 20 20 :44	9C08 49 4E 56 41 4C 49 44 :14
96B0 4F 7E FE F0 2B 14 79 AE :1E	9960 44 52 2E 0D 52 45 41 :C4	9C10 42 52 45 41 4B 0D 4F :0C
96B8 23 A6 28 05 23 23 18 :77	9968 20 20 20 20 20 20 20 :00	9C18 0D 77 8E 03 00 00 00 :15
96C0 F0 23 5E 23 56 EA AF D1 :55	9970 41 44 44 52 2E 0D 4F :F2	9C20 21 00 00 00 00 00 00 :21
96C8 C1 C9 37 18 FA CD C4 08 :6C	9978 44 05 0D 4C 41 42 45 4C :F6	9C28 00 7A 8E 00 00 58 9C :FC
96D0 CD D9 96 67 CD D9 96 6F :4E	9980 20 54 41 42 4C 45 20 41 :E9	9C30 00 00 00 00 00 00 00 :00
96D8 C9 C5 CD EA 96 79 07 07 :62	9988 44 52 2E 0D 4C 41 42 4E :A4	9C38 00 00 00 00 00 38 45 :37
96E0 07 07 47 CD EA 96 78 81 :C8	9990 45 4C 20 53 54 41 52 54 :3F	9C40 37 20 32 31 34 42 39 :43
96E8 C1 C9 CD D2 97 FE 0D 28 :33	9998 20 41 44 44 52 2E 20 :96	9C48 20 20 20 20 20 20 20 :00
96F0 12 F5 CD D2 F8 38 06 4F :63	99A0 20 45 4E 44 20 20 41 44 :B6	9C50 20 4C 44 20 20 20 48 :C4
96F8 F1 C8 CD D2 F8 38 06 4F :63	99B0 44 52 2E 0D 4C 41 42 45 :E5	9C58 2C 39 43 34 42 48 20 :A6
9700 0F 1B E7 CD 14 0F C2 DE :F9	99C0 4C 20 20 45 4E 44 20 20 :A3	9C60 20 20 20 20 00 00 00 :00
9710 19 97 C1 08 9C CD 89 08 C3 :E0	99C8 41 44 44 52 2E 0D 4C 41 :E3	9C68 00 00 00 00 00 00 00 :00
9718 C9 C3 16 9C CD 08 B7 28 FA :90	99D0 42 45 20 53 41 4B 55 53 :45	9C70 00 00 00 00 00 00 00 :00
9720 16 9C CD 89 08 C2 2E 0A 11 :44	99D8 49 43 48 55 0D 43 4F 44 :0C	9C78 00 00 00 00 00 00 00 :00
9728 CD 2E 0A 11 10 9C CD 08 B7 :18	99E0 45 20 20 54 4F 50 20 :E8	9C80 00 00 00 00 00 00 00 :00
9730 08 C3 19 97 0D F7 F1 00 16 :C7	99E8 20 41 44 44 52 2E 0D 4E :C4	9C88 00 00 00 00 00 00 00 :00
9738 C7 06 C7 D3 F7 C6 C7 F1 :0C	99F8 45 4D 4E 4F 4E 49 43 20 :43	9C90 00 00 00 00 00 00 00 :00
9740 01 CF 22 E7 C2 C7 C3 FF :24	99F8 4F 44 45 20 54 45 4E 53 :32	9C98 00 00 00 00 00 00 00 :00
9748 C4 C7 CD FF F0 76 FF F1 :AD	99F8 4F 44 4E 48 55 0D 46 49 :C7	9C98 00 00 00 00 00 00 00 :00
9750 23 F7 09 CF F1 F7 E3 FF :AC	9A00 45 20 4E 41 4D 45 20 4F :F5	9C98 00 00 00 00 00 00 00 :00
9758 E5 FF F9 FF F1 34 FE 46 :45	9A08 20 48 45 59 49 4E 20 53 :13	9C98 00 00 00 00 00 00 00 :00
9760 C7 B6 C7 70 FB F1 21 FF :BD	9A10 45 59 4F 0D 0D 4E 4F 50 :F4	9C98 00 00 00 00 00 00 00 :00
9768 22 F7 36 F1 CB FF F0 63 :5D	9A18 0D 52 4C 43 41 0D 52 52 :E0	9C98 00 00 00 00 00 00 00 :00
9770 F7 70 FE 77 F7 F1 40 C6 :CA	9A20 43 41 0D 52 4C 41 0D 52 :CF	9C98 00 00 00 00 00 00 00 :00
9778 42 C7 44 FF 45 F7 46 FE :BD	9A28 52 41 0D 44 41 0D 43 :B6	9C98 00 00 00 00 00 00 00 :00
9780 5E FF 47 C7 40 E4 F1 F1 :D1	9A30 50 4C 0D 53 43 46 0D 43 :D5	9C98 00 00 00 00 00 00 00 :00
9788 43 C7 F0 0D F7 F1 0D C7 :A9	9A38 43 46 0D 48 41 4C 54 0D :C5	9C98 00 00 00 00 00 00 00 :00
9790 F1 01 CF 22 E7 C2 C7 C3 16 :56	9A40 52 45 54 0D 45 58 58 0D :FA	9C98 00 00 00 00 00 00 00 :00
9798 FF C4 C7 CD FF F0 63 F7 :A6	9A48 44 49 0D 45 49 0D 4E 45 :C8	9C98 00 00 00 00 00 00 00 :00
97A0 F1 F1 F1 21 22 F7 43 :4F	9A50 47 0D 52 45 54 4E 0D 52 :EC	9C98 00 00 00 00 00 00 00 :00
97AB C7 F0 CB FF BC 94 DD FF :AD	9A58 45 54 49 0D 52 52 44 0D :E4	9C98 00 00 00 00 00 00 00 :00
97B0 06 95 ED FF 9B 94 07 C7 :97	9A60 52 4C 44 0D 4C 44 49 0D :D5	9C98 00 00 00 00 00 00 00 :00
97C0 9B 94 76 FF 9B 94 C9 EF :B8	9A68 43 50 49 0D 49 4E 49 0D :D6	9C98 00 00 00 00 00 00 00 :00
97C8 9B 94 F3 F7 9B 94 10 FF :57	9A70 4F 55 54 49 0D 4C 44 44 :C2	9C98 00 00 00 00 00 00 00 :00
97D0 F2 97 18 FF 0D 93 C3 FF :EE	9A78 0D 43 50 44 0D 49 4E 44 :CC	9C98 00 00 00 00 00 00 00 :00
97D8 05 94 CD FF 10 94 EF FF :F1	9A80 0D 4F 55 54 44 0D 4C 44 :E6	9C98 00 00 00 00 00 00 00 :00
97E0 E3 93 08 FF 2A 93 C3 FF :0D	9A88 49 52 0D 43 50 49 52 0D :E3	9C98 00 00 00 00 00 00 00 :00
97E8 41 94 DB FF 4F 94 E3 FF :C7	9A90 49 4E 49 52 0D 4F 54 49 :28	9C98 00 00 00 00 00 00 00 :00
97F0 5D 94 EB FF 68 94 F9 FF :F4	9A98 52 0D 4C 44 44 52 0D 43 :D5	9C98 00 00 00 00 00 00 00 :00
97F8 6D 94 02 EF 65 93 0A EF :33	9AA0 50 44 52 0D 49 4E 44 52 :20	9C98 00 00 00 00 00 00 00 :00
9780 C1 93 20 E7 A6 93 22 EF :A5	9A80 43 0D 4F 54 44 52 0D 42 :AD	9C98 00 00 00 00 00 00 00 :00
9808 CD 93 2A EF D8 93 01 CF :84	9AB8 4C 0D 41 0D 4E 5A 0D 5A :B6	9C98 00 00 00 00 00 00 00 :00
9810 2C 93 03 CF 07 93 08 CF :05	9AC0 0D 4E 43 0D 50 4F 0D 50 :A7	9C98 00 00 00 00 00 00 00 :00
9818 1A 93 C1 CF 1F 93 C5 CF :B3	9AC8 45 0D 50 0D 40 30 30 :69	9C98 00 00 00 00 00 00 00 :00
9820 27 93 09 CF 48 93 04 C7 :18	9AD0 48 0D 32 38 48 0D 31 30 :73	9C98 00 00 00 00 00 00 00 :00
9828 A9 92 05 C7 C3 92 06 C7 :29	9AD8 48 0D 31 38 48 0D 32 30 :77	9C98 00 00 00 00 00 00 00 :00
9830 D8 92 C0 C7 C3 92 06 C7 :29	9AE0 48 0D 32 38 48 0D 33 30 :77	9C98 00 00 00 00 00 00 00 :00
9838 E6 92 C4 C7 92 C7 C7 :17	9AE8 48 0D 33 38 48 0D 42 43 :94	9C98 00 00 00 00 00 00 00 :00
9840 D0 92 C6 C7 77 93 00 C0 :1F	9AF0 0D 44 45 0D 53 50 0D 41 :94	9C98 00 00 00 00 00 00 00 :00
9848 50 92 80 C0 6B 92 00 C0 :1F	9AF8 46 0D 41 44 44 0D 41 44 :AE	9C98 00 00 00 00 00 00 00 :00
9850 D5 94 40 C0 EA 94 80 C0 :1F	9B00 43 0D 53 55 42 0D 53 42 :0C	9C98 00 00 00 00 00 00 00 :00
9858 FC 94 C0 C0 01 95 44 FE :E8	9B08 43 0D 41 4E 44 0D 58 4F :D7	9C98 00 00 00 00 00 00 00 :00
9860 9B 94 4D FF 9B 94 67 F7 :08	9B10 52 4C 43 0D 52 52 43 0D :E2	9C98 00 00 00 00 00 00 00 :00
9868 9B 94 C0 EA 94 94 46 FF :27	9B18 52 4C 43 0D 52 52 43 0D :E2	9C98 00 00 00 00 00 00 00 :00
9870 18 94 56 FF 20 94 5E FF :12	9B20 52 4C 43 0D 52 52 43 0D :E2	9C98 00 00 00 00 00 00 00 :00
9878 25 94 47 FF 78 94 5F FF :61	9B28 41 0D 53 52 41 0D 20 :AE	9C98 00 00 00 00 00 00 00 :00
9880 83 94 4F FF 8B 94 5F FF :E2	9B30 53 52 4C 0D 30 0D 31 0D :79	9C98 00 00 00 00 00 00 00 :00
9888 93 94 42 CF 62 93 43 CF :3F	9B38 32 0D 33 0D 34 35 0D :02	9C98 00 00 00 00 00 00 00 :00
9890 6C 97 4A CF 67 93 48 CF :2C	9B40 36 0D 77 0D 4C 44 0D 4F :73	9C98 00 00 00 00 00 00 00 :00
9898 34 93 40 C7 FC 92 41 C7 :6A	9B48 55 54 0D 49 4E 43 0D 44 :E1	9C98 00 00 00 00 00 00 00 :00
98A0 97 92 0D 07 F7 1F 17 :9C	9B50 45 43 0D 52 45 54 0D 52 :DF	9C98 00 00 00 00 00 00 00 :00
98A8 2F 37 3F 76 C9 D9 F3 FB :AB	9B58 53 54 0D 44 50 0D 43 41 :DF	9C98 00 00 00 00 00 00 00 :00
98B0 44 45 4D 67 6F A0 A1 A2 :8F	9B60 4C 4C 0D 49 4E 0D 50 4F :E8	9C98 00 00 00 00 00 00 00 :00
98B8 A3 8B A9 AA AB A0 B1 B2 :5C	9B68 50 0D 50 55 53 48 0D 4A :F4	9C98 00 00 00 00 00 00 00 :00
98C0 B5 8B B9 BA BB AE 9A B0 :91	9B70 52 0D 44 44 4E 5A 0D 49 :E8	9C98 00 00 00 00 00 00 00 :00
98C8 BA 82 BA 9A BA 96 BA 98 :3C	9B78 4D 0D 45 58 0D 49 0D 52 :AC	9C98 00 00 00 00 00 00 00 :00
98D0 9A E1 8B BA BC BA BF :F7	9B80 0D 42 49 54 0D 52 45 53 :E3	9C98 00 00 00 00 00 00 00 :00
98D8 9A C1 9A 8A CA 9A C7 :64	9B88 0D 53 45 54 0D 48 4C 0D :A7	9C98 00 00 00 00 00 00 00 :00
98E0 9A CA C1 CA CC 9A D2 :9E	9B98 29 0D 46 42 0D 28 42 43 :C8	9C98 00 00 00 00 00 00 00 :00
98E8 9A D6 9A DA DE 9A E2 :08	9BA0 53 50 29 0D 28 48 4C 29 :BE	9C98 00 00 00 00 00 00 00 :00
98F0 9A E6 9A EA EE 9A F1 :17	9BA8 0D 28 43 29 0D 41 46 27 :5C	9C98 00 00 00 00 00 00 00 :00
98F8 9A DE 9A FA EE 9A F1 :1A	9BB0 0D 14 20 0D 49 58 0D 28 :24	9C98 00 00 00 00 00 00 00 :00
9900 9A DE 9B F7 9A 95 9B 9A :6E	9BB8 49 58 28 31 30 48 29 0D :AB	9C98 00 00 00 00 00 00 00 :00
9908 9B DE 9B 9A FA 9A FE :FA	9BC0 46 46 48 0D 28 34 41 48 :C6	9C98 00 00 00 00 00 00 00 :00
9910 9A 02 9B 06 9B 0A 9B 0E :B8	9BC8 29 0D 39 47 34 42 48 0D :7D	9C98 00 00 00 00 00 00 00 :00
9918 9B 12 9B 15 9B 18 9B 1C :C7	9BD0 28 39 43 33 76 48 29 0D :B8	9C98 00 00 00 00 00 00 00 :00
9920 9B 20 9B 23 9B 26 9B 2A :FF	9BD8 38 45 36 30 48 0D 48 4C :CC	9C98 00 00 00 00 00 00 00 :00
9928 9B 2E 9B 30 9B 34 9B 3A :34	9BE0 0D 28 48 4C 29 0D 30 48 :77	9C98 00 00 00 00 00 00 00 :00
9930 9B 38 9B 3A 9B 3C 9B 3E :58	9BE8 29 0D 2A 20 0D 44 41 :32	9C98 00 00 00 00 00 00 00 :00
9938 9B 40 9B 42 9B 44 9B 46 :33	9BF0 54 41 20 45 52 52 4F 52 :3F	9C98 00 00 00 00 00 00 00 :00
9940 5A 2D 38 20 20 20 20 :E3		9C98 00 00 00 00 00 00 00 :00

ラベル付逆アセンブラ メモリ・ダンプ

9EAB 30 34 30 29 29 28 A4 AB :5D	9EEB 58 3D 58 2D 4D 0D 09 6D :EA	9F28 37 00 80 8B 20 53 43 8B :83
9EB0 24 45 30 34 31 29 29 29 :79	9EF0 34 00 80 8B 20 81 58 2C :94	9F30 43 30 20 80 8C 20 48 3D :44
9EB8 0D C3 6C 30 00 80 80 0D :79	9EF8 32 29 3D 22 94 22 20 20 :B0	9F38 35 30 30 3A 80 8D 20 20 :1C
9EC0 CA 6C 31 00 80 80 0D 04 :4B	9F00 80 8C 20 36 34 0D 20 6D :30	9F40 35 39 0D 63 6D 3B 00 4C :CF
9ECB 6C 32 00 58 3D 58 2B 4D :03	9F08 35 00 80 8B 20 81 58 2C :95	9F48 3D 4C 2D 2E 31 3A 46 4D :E2
9ED0 0D F1 6C 33 00 80 8B 20 :CB	9F10 32 29 3D 22 86 22 20 80 :02	9F50 3D 46 4D 2B 4B 4C 3A 4B :11
9ED8 28 58 3D 39 29 28 28 58 :CA	9F18 8C 20 37 35 0D 29 6D 36 :F1	9F58 4C 3D 30 3A 4B 3D 30 1B :C3
9EE0 3D 32 35 29 20 80 8C 20 :19	9F20 00 80 90 20 4B 0D 46 6D :3B	

●MZ-80K/C版のシステム・プログラムをMZ-80Bで!

SELF RELOCATABLE DEBUGGER

MZ-80B (マシン語) 

セルフ・リロケートブル・デバッガ

COMPAC

MZ-80K/C用のマシン語プログラム開発システム「セルフ・リロケートブル・デバッガ」(逆アセンブラを含む)をMZ-80Bに移植したので紹介します。

コマンドの説明

(1) M コマンド (メモリ・ダンプ)

コマンドの入力待ちに、**[M]** をキーインして、16進でダンプのスタート番地とエンド番地をキーインします。

ダンプの途中で **[スペース]** を押すと、ストップしてエディット・モードになります (カーソル・エディット可能)。エディット・モードで **[SHIFT BREAK]** か **[CR]** を押すとダンプを再開します。

また、16進数以外のキーを入力したときも同様です。ダンプ中に、**[CR]** を押すと、『コマンド待ち』に戻ります。

また、すべてのコマンドで16進数入力中に **[CR]** キーを押すと、コマンド待ちに戻ります。

(2) W コマンド (メモリ・ライト)

入力中、**[←]** キーを押すと改行して1バイト前に戻ります。**[CR]**、**[DEL]** キーでコマンド待ちに戻ります。

(3) G コマンド (プログラム実行)

アドレスを入力するとそのアドレスへジャンプします。

(4) C コマンド (サブルーチン・コール)

実行後はコマンド待ちになります。

(5) H コマンド (ホーム・クリア)

画面をクリアしてカーソルをHOMEに持っていきます。

(6) ! コマンド (ジャンプ・モニタ)

モニタに戻ります。

(7) S コマンド (セーブ)

セーブするプログラムのファイル・ネーム、先頭アドレス、末尾アドレス、オートスタート・アドレスを決定でき

ます。

(8) L コマンド (ロード)

ファイル・ネームを入力するとLOADします。ファイル・ネームを入力しないときは、プログラムを見つけることにLOADするかを聞いてきます。

(9) V コマンド (ベリファイ)

最初に見つけたプログラムとのベリファイをします。

(10) P コマンド (ASCII)

ASCIIコードを指定番地以降に書き込みます。1行入力して **[CR]** すると、書いたものを16進表示してエディットモードになります。

あとは、**M** コマンドと同じです。

(11) I コマンド

ある領域のメモリの内容を同一のコードで塗りつぶしてしまうコマンドです。

(12) X コマンド (トランスファ)

転送前後の領域が重なってもかまいません。

(13) O コマンド (カルキュレート)

16進4桁の加減算をするコマンドです。

(14) T コマンド (ASCIIダンプ)

メモリの内容をASCIIコードで、ダンプするコマンドです。

(15) B コマンド (ブレイク・ポイント)

ブレイク・ポイントを設定できます。ブレイク・カウン트는1回だけです。ブレイク・カウン트를0にすると、そのブレイク・ポイントは解除されたことになります。また、アドレスを書き換えるとそれ以前のブレイク・ポイントは自動的に解除されます。また、同じアドレスにブレイク・ポイントを2つ以上設定しないようにしてください。

ブレイク・ポイントはGコマンドを行なう直前に設定してください。ブレイク・ポイントは非常にシビアなので注意して使ってください。

(16) A コマンド (レジスタ表示)

全レジスタの内容を表示し、その内容の訂正を行なうコマンドです。訂正するときは、まず、訂正しようとするレジスタの名前の表示してある行をまず **[CR]** して入力し、その直後に内容を訂正して **[CR]** します。

(17) N コマンド (リスタート)

レジスタ・バッファの内容をレジスタに入れて、リスタートします。

(18) ? コマンド (画面コピー)

画面をプリンタへ打ち出します。

(19) D コマンド (ディスアセンブル)

領域を指定すると、逆アセンブルします。25行表示すると一時ストップして「コマンド待ち」になり、この状態でのコマンドは4つあります。

? コマンドは画面の内容をプリンタに出力します。それ

が終わると、コマンド待ち (D コマンド内での) になります。次に、R コマンドでディスアセンブルするアドレスを訂正できます。ただし、最初に指定した END アドレス以降を指定すると無視してディスアセンブルを続けていきます。

N コマンドは、逆アセンブルを続行させるコマンドです。また、いかなる状態でも、**[CR]** を入力するとコマンド待ちになります。

(20) R コマンド (リロケート)

リロケートする領域を入力してリロケートする先頭アドレスを入力します。次に領域内のデータの部分を指定します。自分自身をリロケートするには次のようにします。

```
FROM? B000 TO? C3AE TOP? 2000
DATA WREA? BF4F-C3AE
```

□参考文献

ロリコン・クラブ: "SELF RELOCATABLE DEBUGGER": 1/0, '80年11月号

SELF RELOCATABLE DEBUGGER プログラム・リスト

```
B000 CD EE 0C 11 DB C1 CD B6 13F
B008 05 CD 2E 0A 3E 5D CD C6 138
B010 08 CD 14 0F CD DB B6 FE 156
B018 57 CA B6 80 FE 4D CA AF 11B
B020 80 FE 47 CA 24 B1 FE 58 1EA
B028 CA 71 B2 FE 41 CA 1A B8 1CB
B030 FE 44 C6 61 BC FE 4E CA 13F
B038 F7 B7 FE 3F CA D4 BA FE 1C9
B040 53 CA 2E B1 FE 52 CA BA 1D0
B048 B8 FE 42 CA 52 B5 FE 43 1A
B050 CA A5 B1 FE 48 CA B6 B1 197
B058 FE 50 CA B7 B4 FE 4F CA 1AA
B060 B8 B4 FE FF CA D4 B3 FE 1B8
B068 56 CA CD B2 FE 54 CA 4A 105
B070 B3 FE 49 CA 96 B3 CD AC 157
B078 CA BE B1 FE 21 CA 00 0D 122
B080 CD 14 0F CD C3 09 B0 CD C6 1FF
B088 08 CD 79 B8 CA 09 B0 06 192
B090 08 CD 2E 0A CD 47 B8 CD 1A9
B098 B9 B8 28 06 77 23 10 F7 113
B0A0 1B ED FE 04 C2 09 B0 3E 1C0
B0A8 5F CD C6 08 2B 18 E0 CD 1EA
B0B0 C6 08 CD 79 B8 CA 09 B0 152
B0B8 EB CD 79 B8 CA 09 B0 13C
B0C0 1E B1 DA 09 B0 EB CD 2E 140
B0C8 0A CD 08 05 06 08 7E 100
B0D0 50 B8 CD 1E B1 2A 13 23 0A6
B0D8 10 F4 CD 32 08 FE 0D CA 1E0
B0E0 09 B0 FE 20 50 CD F1 195
B0E8 B0 18 DB CD F1 B0 C3 09 100
B0F0 B0 C5 D5 E5 CD 2E 0A 11 145
B0F8 7C CD C4 A6 CD 14 06 190
B100 38 1B 13 13 13 0E 08 1B2
B108 13 D5 CD 23 06 D1 3B 0A 1F1
B110 77 23 13 13 0D 20 F1 C3 1A1
B118 F7 B0 E1 D1 C1 C9 72 92 1F1
B120 C0 7D 93 C9 CD C6 08 101
B128 79 B8 CA 09 B0 E9 CD C6 133
B130 08 CD 2E 0A 11 FE C1 CD 1AA
B138 09 B0 CD 79 B8 CA 09 B0 115
B140 22 54 11 E3 CD A4 0A 0E 1D2
B148 09 B0 CD 79 B8 D1 CA 09 136
B150 B0 CD 1E B1 DA 09 B0 AF 1BE
B158 ED 52 23 22 11 00 00 0E7
B160 00 11 09 CD C2 B6 05 CD 131
B168 79 B8 CA 09 B0 22 56 11 140
B170 00 00 00 11 12 CD B6 168
B178 05 11 7C C3 CD A4 0A 0E 1D2
B180 08 13 10 FD 21 41 11 EB 0B9
B188 01 03 00 B0 21 00 00 1CF
B190 22 08 11 21 F0 00 22 0A 17D
B198 11 22 0C 11 CD 51 02 CD 138
B1A0 B2 02 C3 09 B0 18 C6 0E 19B
B1A8 CD 79 B8 CA 09 B0 11 B3 148
B1B0 B1 D5 E9 C3 09 B0 3E 06 12F
B1B8 CD C6 08 C3 00 B0 CD C6 1A1
B1C0 08 CD C4 08 11 7C C3 CD 1BE
B1C8 CA 06 13 13 1A FE 0A 1D2
B1D0 C2 1C B2 CD BE 02 DA 08 1D2
B1D8 B2 CD 3A B2 CD 2E 0A 11 1B1
B1E0 27 C2 CD 89 08 CD C4 08 1E0
B1E8 CD D4 B3 20 E6 CD 2E 0A 15F
B1F0 11 2D 02 CD 89 08 11 41 1F0
B1F8 11 21 10 00 19 36 0D CD 168
```

```
B200 B9 08 CD B2 02 DA 08 B2 1A9
B208 C3 09 B0 FE 02 CA 0B 00 1FF
B210 CD 2E 0A 11 49 CD CD 89 177
B218 08 C3 09 B0 D5 CD BE 02 1B6
B220 DA 09 B0 CD 3A B2 D1 D5 1F4
B228 21 41 11 06 10 CD 3A 0A 196
B230 C4 D0 B3 C2 1D B2 D1 0A 113
B238 F0 B1 CD 2E 0A 11 41 02 1FA
B240 CD 89 08 11 41 11 21 10 1F2
B248 00 09 36 0D CD 89 08 CD 1B7
B250 2E 0A 11 1E C2 CD 89 08 CD 1E7
B258 2A 54 11 CD 42 BB 11 23 1B0
B260 C2 CD 89 08 ED 58 52 11 1CB
B268 19 2B CD 42 BB CD 2E 0A 113
B270 C9 CD C6 08 11 FE C1 CD 101
B278 B6 05 CD 79 B8 CA 09 B0 13F
B280 E5 11 04 C2 CD 89 08 CD 1E7
B288 79 B8 D1 CA 09 B0 CD 1E 173
B290 B1 DA 09 B0 AF ED 52 23 155
B298 ED 53 41 C3 22 43 C3 11 17D
B2A0 2D C2 CD 89 08 79 B8 14E
B2A8 CA 09 B0 ED 58 41 C3 CD 19C
B2B0 1E B1 D2 B2 B2 ED 13 155
B2B8 43 C3 ED B0 C3 09 B0 ED 10C
B2C0 4B 43 C3 09 2B EB 09 2B 1A4
B2C8 ED B8 C3 09 B0 CD C6 08 1BC
B2D0 C4 08 11 7C C3 CD A4 15A
B2D8 06 13 13 13 1A FE 0D C2 126
B2E0 23 B3 CD BE 02 DA 09 B0 1C6
B2E8 CD 3A B2 CD 29 0A 00 0B9
B2F0 00 00 00 00 00 00 00 000
B2F8 00 00 00 00 00 29 0A 11 111
B300 38 C2 CD 89 08 11 41 11 1BE
B308 21 10 00 19 36 0D CD 89 1E3
B310 08 CD BE 02 DA BE 3D CD 120
B318 29 0A 11 46 C2 CD 89 08 1A0
B320 C3 09 B0 D5 CD BE 02 DA 1B8
B328 0E B2 CD 89 08 D1 D5 21 140
B330 41 11 06 10 CD 3A 0A C2 137
B338 24 B3 D1 C3 FE B2 CD 29 10F
B340 0A 11 49 C2 CD 89 08 C3 147
B348 09 B0 CD C6 08 CD 79 B8 155
B350 CA 09 B0 EB CD 79 B8 CA 139
B358 09 B0 CD 1E B1 DA 09 B0 1E0
B368 05 CD 2E 0A CD D8 05 0A 1B0
B368 05 CD 0A 10 F0 06 10 1BF
B370 7E FE 1D 0C CA 08 D4 C6 1DB
B378 08 00 CD 1E B1 CA 09 1027
B380 23 10 ED CD 32 0E FE 0D 132
B388 CA 09 B0 DB EA FE FD 2B 168
B390 FA C3 61 B3 C6 08 CD C6 132
B398 08 11 FE C1 CD B6 05 CD 120
B3A0 79 B8 CA 09 B0 E5 11 04 1B1
B3A8 C2 CD 89 08 CD 79 B8 D1 1F2
B3B0 CA 09 B0 CD 1E B1 DA 09 102
B3B8 B0 EB D5 11 4F C2 CD 89 1E8
B3C0 08 CD 89 B8 D1 CA 09 B0 16D
B3C8 47 70 CD 1E B1 CA 09 B0 1D6
B3D0 23 C3 09 B3 CD CE 04 CD 1CE
B3D8 B8 BB FE 57 CB 08 CD B1 178
B3E0 C4 CD C6 0A 08 C9 00 174
B3E8 00 00 00 00 00 00 00 000
B3F0 00 00 00 00 00 00 00 000
B3F8 22 40 C3 ED 43 49 C3 ED 158
```

```
B400 53 4B C3 F5 E1 22 47 C3 163
B408 2A 71 11 22 60 C3 E1 E5 1B7
B410 3E C5 CD CD CD 02 D2 B8 1B3
B418 3A 4F C3 FE 01 CA 3B 8A 121
B420 CD 1B 0F 00 FE 43 20 07 3E 1BE
B428 01 32 4F C3 1B 0A FE 21 1A6
B430 CA 5A B4 FE 11 C2 20 B4 17D
B438 CD 1B 0F 0E 21 CA 5A B4 10F
B440 2A 60 C3 22 71 11 2A 47 162
B448 C3 E5 F1 2A 4D C3 ED 4B 10B
B450 49 C3 ED 5B 48 CD C3 65 194
B458 B4 C9 E1 21 92 03 22 39 16F
B460 10 F3 C3 09 B0 F3 F5 14C
B468 3E 74 32 07 E0 3E B0 32 1EB
B470 07 E0 21 06 E0 36 01 36 15B
B478 00 2B 36 02 36 00 3E 13 1EA
B480 3D 20 FD E1 F1 FB C9 CD 1B0
B488 B4 C6 08 CD 79 B8 CA 09 B0 152
B490 22 62 C3 00 00 00 11 5A 1AE
B498 C2 CD B6 05 11 7C C3 CD 167
B4A0 A4 06 06 06 13 10 FD 1A 10F
B4A8 77 FE 0D 28 04 23 17 1B 1FC
B4B0 F6 EB 2A 62 C3 C3 C6 B0 169
B4B8 01 3D 02 C3 11 5D C2 CD 89 121
B4C0 08 CD 2E 0A 3E 2A CD C6 102
B4C8 08 CD 7E B8 CA 09 B0 22 183
B4D0 64 C3 CD C4 08 CD DB B8 120
B4D8 00 00 00 FE 28 2B 0F FE 158
B4E0 2D 2B 0C CD 14 0F 1B ED 15E
B4E8 CD C6 08 3E 01 1B 05 CD 1C4
B4F0 C6 08 3E 02 32 6E C3 CD 13E
B4F8 CA 08 3E 24 C6 08 CD 08 196
B500 7E B8 CA 09 B0 3A 6E C3 127
B508 FE 01 20 14 ED 5B 6A C3 1A2
B510 19 DA 49 85 11 70 C2 CD 101
B518 B9 08 CD 47 B8 C3 C1 B4 198
B520 ED 5B 64 C3 EB CD 1E B1 176
B528 3B 0F AF ED 52 11 70 C2 105
B530 CD 89 08 CD 47 B8 C3 C1 1B1
B538 B4 EB AF ED 52 11 75 C2 1B5
B540 CD 89 08 CD 47 B8 C3 C1 1B1
B548 B4 11 7B C2 CD 89 08 C3 123
B550 C1 B4 CD C6 08 3E C3 32 1A3
B558 3B 0F AF ED 52 11 70 C2 105
B560 00 00 01 84 C2 CD B6 1DA
B568 05 11 70 C3 21 6A C3 36 1CD
B570 31 0E 03 CD 2E 0A 3A 6A 1EB
B578 C3 CD C6 08 CD C4 08 13 10A
B580 1A 6F 13 1A 47 CD 47 BB 1EC
B588 CD C4 08 13 1A 6A 0F 1AB
B590 0F 0F 0F 0F 0F 0F 0F 0F 19A
B598 C3 3C 32 6A C3 13 0D C2 40
B600 73 B5 11 7C C3 CD A4 0E 1EF
B608 1A 06 30 FE 01 CA B0 B5 15B
B610 FE 02 CA C5 B5 FE 03 CA 10F
B618 CD B5 C3 09 B0 3E 00 32 16E
B620 C5 C3 C3 D2 B5 3E 04 32 1ED
B628 C6 C3 C3 D2 B5 3E 08 32 1F1
B630 C6 C3 13 13 CD 14 06 DA 116
B638 09 B0 22 66 C3 06 04 13 121
B640 10 FD 3E 30 12 CD 23 06 1B3
B648 DA 09 B0 32 6D C3 11 70 176
B650 C3 26 0A 6C C3 AF 19 1DA
B658 22 6B C3 23 5E 23 56 2A 171
```

```

B600 66 C3 CD 0E B1 C2 42 B6 17F
B608 2A 6B C3 06 03 25 10 FD 18E
B610 3A 6D C3 47 C8 27 CB 27 195
B618 CB 27 CD 0B 77 3A 6D 182
B620 C3 FE 00 C4 2B 6C DE 168
B628 0A C3 02 0A 0A 0A 0A 1F7
B630 2A 64 C3 77 3E 00 2A 68 19A
B638 C3 06 03 77 23 10 FC C3 135
B640 2B 66 3A 6D C3 FE 0A C4 10E
B648 2B 66 ED 5B 68 C3 13 1A 17C
B650 AF 13 13 10 EA 11 70 C3 172
B658 ED 5B 68 C3 2A 66 C3 7E 144
B660 12 3E FF 77 13 EB 78 77 186
B668 23 7A 77 C3 08 B6 22 40 104
B670 C3 ED 43 49 C3 ED 53 48 18A
B678 C3 F5 E1 22 C3 03 22 C4
B680 5B C3 FD 22 5D C3 ED 57 1A1
B688 22 5F C3 03 75 59 C3 E1 181
B690 2B 22 57 C3 E5 09 22 55 19C
B698 C3 ED 47 51 C3 03 53 19A
B6A0 C3 09 08 51 22 4F C3 16E
B6A8 0B 2A 59 C3 23 22 59 10F
B6B0 C3 11 70 C3 13 06 03 1A 13D
B6B8 AF 13 1A 67 05 ED 5B 57 177
B6C0 C3 CD 1E B1 C1 6A B6 14A
B6C8 13 13 10 EA 11 70 C3 172
B6D0 13 06 03 1A 6F 13 1A 67 139
B6D8 3E FF 77 13 13 10 F3 10F
B6E0 E1 23 25 2A 47 C3 E5 F1 1F3
B6E8 ED 4B 48 C3 ED 5B 68 C3 172
B6F0 2A 4D C3 C9 13 EB 7E 47 18C
B6F8 E6 0F 3D 57 F8 6A 70 47 18C
B700 F1 F5 80 77 F1 28 0B 2B 12C
B708 2B 28 7A 2A 57 C3 77 C3 14E
B710 4B 7E 2A 57 C3 77 C3 14E
B718 OF 0F 70 77 C3 23 B7 E1 190
B720 C3 09 80 11 91 C2 00 00 1E0
B728 00 CD B6 05 CD 2E 0A 11 19E
B730 47 C3 06 04 CD 6E 87 11 117
B738 ED 52 00 00 00 00 00 00 00
B740 CD 2E 0A 11 4F C3 06 04 13C
B748 CD 6E 87 11 C2 02 00 00 187
B750 00 CD B6 05 CD 2E 0A 11 19E
B758 57 C3 06 04 CD 6E 87 11 117
B760 1A 67 CD 42 B6 C3 0A 1A 68
B768 F3 1A CD 50 B8 C3 05 C3 186
B770 1A CD 50 B8 1B 1A CD 50 144
B778 B8 13 C3 11 10 F0 C9 CD 13B
B780 C6 08 11 70 C3 13 06 03 12E
B788 1A 6F 13 1A 67 C3 23 B7 E1 190
B790 57 C3 CD 1E B1 C1 2B 3F 18E
B798 13 13 10 EB 08 2A 4F 185
B7A0 C3 E5 F1 08 D9 6D 4B 51 103
B7A8 C3 ED 5B 53 C3 2A 55 C3 163
B7B0 D9 ED 7B 39 C3 0A 0A 0A 1F7
B7B8 C3 FD 2A 5D C3 3A 5F C3 166
B7C0 ED 47 2A 47 C3 E5 F1 ED 12B
B7C8 4B 49 C3 ED 5B 48 C3 2A 107
B7D0 57 C3 E5 2A 47 C3 09 1B 11D
B7D8 1B 1A 2A 57 C3 77 08 2A 122
B7E0 4F C3 E5 F1 08 D9 ED 4B 101
B7E8 51 C3 ED 5B 53 C3 2A 55 101
B7F0 C3 09 ED 7B 39 C3 0A 2A 127
B7F8 5B C3 FD 2A 5D C3 3A 5F C3 166
B800 C3 ED 47 2A 47 C3 E5 F1 101
B808 ED 4B 49 C3 ED 5B 48 C3 19A
B810 2A 57 C3 E5 2A 47 C3 2A 122
B818 4B 48 C4 C6 08 CD 2E 0A 19A
B820 CD 23 87 11 7C C3 0A 1A 68
B828 06 13 1A FE 50 C4 B6 8B 189
B830 FE 41 2B 03 C3 09 80 13 1F9
B838 1A FE 20 2B 0F FE 27 28 18C
B840 C3 C3 09 80 21 47 C3 23 192
B848 44 C3 18 06 21 47 C3 23 192
B850 44 C3 11 7C C3 0A 0A 1E
B858 13 06 04 03 CD 23 06 D1 189
B860 0A 09 80 67 13 13 13 05 10E
B868 CD 23 06 D1 189 80 85 1F2
B870 13 13 13 EB E5 2A 64 C3 15A
B878 77 23 72 23 22 64 C3 E1 155
B880 ED 10 D8 C3 23 B8 11 7C 1FE
B888 C3 0A 46 21 57 C3 22 197
B890 44 C3 13 06 04 03 CD 23 06 D1 189
B898 0A 09 80 EB E5 2A 64 C3 15A
B8A0 73 23 72 23 22 64 C3 E1 155
B8AB ED 13 13 13 13 13 10 E5 13F
B8BC CD 23 2A 64 C3 77 C3 181
B8C8 23 8B CD 2A 08 C4 0A 10F
B8C0 11 D9 C2 CD 89 08 CD 2E 105
B8CB 0A 11 FE C1 CD 89 08 CD 105
B8D0 79 B8 C4 09 80 EB D5 11 188
B8DB 04 C2 CD 89 08 D1 CD 79 13B
B8DC 0B 7A C2 CD 13 B1 6D 18E
B8E8 09 80 ED 53 C3 23 25 14E
B8F0 C3 AF ED 52 22 37 C3 11 1DE
B8F8 20 C2 CD 89 08 CD 79 88 14E
B900 C4 09 80 22 C3 ED 5E 169
B908 57 C3 EB ED 1B DA 2A 17B
B910 89 2A 37 C3 1A 22 5B 33 171
B918 C3 CD 1E B1 DA ED 5B C3 107
B920 09 80 CD 2E 0A 11 E2 173
B928 CD 89 08 CD 79 B8 C4 68 191
B930 11 D9 C2 CD 89 08 EB 122

```

```

B938 CD 79 B8 C4 68 B9 CD 1E 107
B940 B1 DA 68 B9 ED 53 3B C3 4EA
B948 22 3D C3 ED 5B 33 C3 2A 18A
B950 3B C3 CD 1E B1 DA 68 B9 195
B958 ED 5B 3D C3 2A 35 C3 CD 137
B960 1E B1 DA 68 B9 C3 7A B9 1C0
B968 CD 2E 0A 11 F0 C2 CD 89 11E
B970 08 21 FF FF 22 3B C3 22 169
B978 3D C3 CD 2E 0A 11 FE C2 106
B980 CD 89 08 CD 5B 33 FE 59 115
B988 1E B1 DA 68 B9 C3 7A B9 1C0
B990 2A 39 C3 ED 5B 33 C3 AF 113
B998 ED 52 22 3F C3 ED 5B 33 10E
B9A0 C3 2A 39 C3 E5 CD 46 B8 19B
B9AB E1 DA B3 B9 1A 77 13 23 1E
B9B0 C3 3B BA 05 CD 64 BA 15C
B9B8 E1 D1 7B FE 01 CA CD 89 179
B9C0 FE 02 CA CD 89 FE 03 CA 11B
B9CB D6 89 C3 1B 8A 1A 77 13 1CB
B9D0 23 10 FA C3 3B 1A 1A FE 1FA
B9E0 DD 2B FE 02 FE 2B FE 77 17F
B9E8 13 23 EB D5 E5 23 5A ED 18A
B9F0 E5 64 C3 2B E5 CD 56 BA 167
B9F8 E1 D1 EB DA 1A BA E5 D5 101
B9A8 ED 5B 3F C3 2A 64 1A
B9B8 22 64 C3 D1 E1 D5 ED 5B 11E
B9C8 64 C3 73 23 72 2B D1 23 148
B9D0 23 13 13 C3 3B BA 06 02 106
B9E0 1A B3 CD 09 13 1A 1B FE CB 15A
B9F0 C3 BA FE 3A C3 0A 11 13 10F
B9A0 02 1A 77 13 23 10 FA 1D9
B9B0 C3 E2 B9 06 04 C3 CD 89 1B1
B9C0 E5 2A 35 C3 CD 1E B1 E1 18A
B9D0 09 80 B0 C3 A4 B9 2A 30 11A
B9E0 CD 1E B1 D8 2A 3B C3 157
B9F0 CD 1E B1 EB 2A 3C C3 157
B9A0 EB CD 1E B1 EB CD EB 06 12C
B9B0 01 11 2C C3 CD EB BA 26 168
B9C0 ED 79 EB CD 1E B1 EB 2A 3B C3 157
B9D0 2B 09 FE FD 2B 25 21 187
B9E0 02 C3 1B 25 04 13 1A FE 131
B9F0 4E CA 1A BA FE 56 CA A1 12A
B9A0 BA FE CA A1 BA FE 72 1AB
B9B0 BA FE A1 BA FE 72 1AB
B9C0 BA FE 1B 13 04 13 21 17 135
B9D0 BA FE C4 BA FE F0 2B 07 12B
B9E0 FE 05 38 01 04 04 C9 186
B9F0 7E 13 1A FE F0 2B 07 12B
B9A0 09 CB 1B FA 1E 0F 18E
B9B0 7E FE F0 C4 BA AE 23 12E
B9C0 A6 20 F4 C9 CD C6 08 CD 1EB
B9D0 03 C3 09 B0 21 00 00 00 103
B9E0 1A 19 CD BA 1E 1B 0F 18E
B9F0 02 05 CD C9 CD 0A 1E3
B9A0 BB 23 1D 20 F2 CD FC BA 10C
B9B0 15 20 EA 19 CD 0A CD 06 103
B9C0 BB CD 0A BC C9 00 F5 3A 134
B9D0 08 08 C3 0A F1 D7 FF 3E 144
B9E0 03 FE 3E 01 CD 1C BB 13A
B9F0 AF 03 FE C9 D9 57 01 00 17A
B9A0 08 08 FE E6 0A B0 02 1AB
B9B0 09 C9 1E 10 1D 20 F8 155
B9C0 BB B1 20 D9 CD 2E 0A 11A
B9D0 11 49 C2 CD 89 08 F1 F1 15C
B9E0 F1 C9 F5 CD C4 08 F1 CD 106
B9F0 08 05 C9 70 CD 55 BB C9 1C9
B9A0 05 CD C4 08 F1 CD 05 152
B9B0 08 08 CB CF 87 EB 08 118
B9C0 FE 09 F0 AF C9 30 C6 16 18B
B9D0 30 CD 12 00 C9 C6 37 18 ED
B9E0 FB F5 CD C4 08 F1 CD BE 1D2
B9F0 BB CB 67 CD BE BB CF 337
B9A0 C9 F5 CD C4 08 F1 CD A4 89
B9B0 BB CB 07 07 07 C5 47 18E
B9C0 AA BB 22 80 04 C1 1CB
B9D0 CD C4 08 F1 CD 08 08 10C
B9E0 CD C4 08 F1 CD 08 08 10C
B9F0 F5 FE 30 3B 1D FE 3A 30 1E0
B9A0 CD C6 08 F1 D6 30 1B 1B2
B9B0 0E FE 41 3B 0D FE 47 30 107
B9C0 CD C4 08 F1 D7 FF 3E 144
B9D0 F0 C9 F1 CD 14 OF 1B C2 17E
B9E0 D5 E5 CD 32 08 FE 0A C4 89
B9F0 DA BB E1 CD 19 00 87 20 17E
B9A0 47 1A BC CD 19 00 87 20 17E
B9B0 2B F7 CD 49 BC 36 05 F5 1C5
B9C0 11 00 00 1B 7A B3 02 06 181
B9D0 F1 E1 D1 C9 3E 0D 06 F8 191
B9E0 BB C9 32 70 11 C9 AF 32 1E1
B9F0 0E 20 27 C1 D7 2E 32 171
B9A0 0E 20 27 C1 D7 2E 32 171
B9B0 13 7E OF D8 3E CF 3A 1AC
B9C0 BC EB CD 49 BC FF 7E 7E 159
B9D0 E1 77 79 7E OF D8 3E 1AC
B9E0 BC F5 39 CD E0 77 7E 159
B9F0 CA FR 3A 02 E0 77 3B FA 17C
B9A0 C9 2A 71 11 C5 D5 E5 C1 185
B9B0 11 2B 00 21 D8 CF 19 05 1F1
B9C0 2F 5B CB 0E 00 00 D1 C1 1A5
B9D0 CD C4 08 F1 D7 FF 3E 144
B9E0 08 08 CD 89 79 B8 C4 09 17B

```

```

B9C0 80 EB 3E 00 32 48 C3 D5 10E
B9CB 06 20 3E 00 11 7C C3 12 103
B9CD 13 10 FC D1 E5 D5 CD 0E 185
B9CE 80 13 01 01 CD 8C 05 00 00 FD
B9CF 00 0E 0A 3A 06 C1 47 7E 1A4
B9D0 00 50 89 23 0C 10 F5 14D
B9DA 05 91 47 3E 01 C6 03 10 1F5
B9DB CF D6 03 47 CD C4 08 10 1C5
B9DC FB D5 11 7C C3 89 08 08 17E
B9DD 01 CD 1E B1 81 09 80 CD 1C5
B9DE 32 02 08 F5 07 19 FE 168
B9DF 0A 09 80 3A 6B C3 32 159
B9E0 48 C3 FE 18 DA 77 BC 1C 11E
B9EB 2E 0A CD 8B 08 FE 20 C4 180
B9EC 02 8C 5E 52 28 10 FE 0D 1C1
B9ED 0A 09 80 09 80 3F 25 0C 144
B9EE DD BA D1 E1 1E 84 3E 52 105
B9EF CD C6 08 CD 79 8B 2B 1A9
B9F0 05 CD 1E B1 3B 04 E1 C3 181
B9F8 72 BC C1 C3 72 8C 22 02 104
B9D10 C1 9B CD 6A 7B 32 D6 119
B9D18 C1 E7 32 04 C1 CD C7 8D 170
B9D20 FE CB 2B 0E FE ED 2B 12 124
B9D28 FE DD 2B 16 FE ED 2B 16 152
B9D30 1B 2A CD 27 02 11 D1 10D
B9D38 1B 26 CD C7 8D 20 19 1C 18A
B9D40 1E 3E 16 18 02 3E 18 1FA
B9D48 32 04 C1 CD C7 8D FE CB 1E1
B9D50 2B 01 21 27 00 21 AB C1 1FE
B9D58 CD B1 20 00 00 00 00 00 00
B9D60 CD D4 B8 CB 07 CD 90 B0 187
B9D68 CD 0A BE C9 21 CE BF CD 103
B9D70 1D BF C9 CD C7 8D CD C7 18A
B9D78 B0 21 DD CD 00 CD 8B 111
B9D80 0B 08 1E 18 1E 18 1E 18 1E
B9D88 53 02 C1 CD 90 18 0B 1D 1F0
B9D90 5F CD 1D BF 7E E6 3F CD 170
B9D98 16 00 F5 3E 20 CD 49 BF 13E
B9DA0 F1 21 4F 20 19 5E 23 103
B9DA8 7E 78 CD B7 8D F1 FE 151
B9DB0 00 8C 16 2C CD 25 BF 21 1DC
B9DB8 09 BE D5 16 00 2F 19 D1 1FB
B9DC0 79 1F 1F 1F E6 07 E9 D9 185
B9DD0 7E F3 7E 02 D2 C1 C1 103
B9DE0 7E D9 49 07 97 BE 2B CD 1F9
B9DF0 79 BE 23 3E 00 CB 8E 7E 1A9
B9E00 23 CD C4 BA FE F0 2B 07 12B
B9E08 1A BE 23 7E 23 2B 00 CD 145
B9E10 7E 20 F3 7E 23 2B 0E 15F
B9E18 20 F9 37 C9 CB 7E CB 23 1AD
B9E20 23 23 1B FB 3E 0D C3 49 1AD
B9E28 BF C9 CD C7 8D CD 2B BF 190
B9E30 5E 4B 1B C3 C3 C7 8D 73
B9E38 CD C7 8D C7 8D BF 7B CD 158
B9E40 36 BF 1B EC 79 E6 3B 18 1AB
B9E48 E4 3E 49 C9 29 2B 02 3E 17B
B9E50 52 18 24 C3 3F BF 2B 01 1F7
B9E58 30 C3 47 BF 00 00 00 00 00
B9E60 00 CD 23 BF CD 5A BE 194
B9E68 18 47 CD 23 BF CD 77 BE 10
B9E70 3F CD 5E BE 27 C3 68 18E
B9E78 49 BF 3E 0E 18 02 3E 20 1CC
B9E80 02 3E 0E 18 02 3E 20 1CC
B9E88 1B 02 3E 0E 18 02 3E 20 1CC
B9E90 08 BF FE 0A 2B E7 E6 183
B9E98 3A C6 08 FE 0C 02 08 BF 19F
B9EA0 3A C1 C6 0C 08 BF 8F 12B
B9EA8 FE 30 C6 10 1B 5A CD C7 166
B9EB0 FE 30 C6 10 1B 5A CD C7 166
B9EB8 3E 2B CD 49 BF F1 C3 0D 1FF
B9EC0 BE FE FF 2B 12 FE 2B 19
B9EC8 12 47 3E FE 0F 35 3E 2D 185
B9ED0 CD 49 47 3E FE 0F 35 3E 2D 185
B9ED8 01 1B 0C 3E 10 08 D9 79 1FC
B9EE0 EA 07 FE 06 20 22 CD 23 123
B9EE8 BF CD 80 BE 3A D1 C1 87 150
B9EF0 2B 9F CD C7 8D C1 BF 17 155
B9EF8 2B 97 1B 2B 03 BF 1A 10A
B9F00 2D ED 44 CD 25 BF 18 BA 1AD
B9F08 21 A3 BF 85 4F C0 21 24 1CC
B9F10 7E E6 7E FE 20 C4 8F 1CD
B9F18 BE FB 23 1B F3 FD 7C 7E
B9F20 1B 1B 1B 1B 1B 1B 1B 1B
B9F28 CD C9 FE 0A 3B 07 F5 17E
B9F30 3E 30 CD 49 BF F1 F5 0F 13B
B9F38 OF OF CD C7 BF F1 E6 CF 13B
B9F40 OF OF 0A 3B 02 C6 07 C6 1E4
B9F48 CD 49 47 3E FE 0F 35 3E 2D 185
B9F50 00 D6 00 01 00 05 01 6E 1AB
B9F58 00 2A D6 09 00 61 6E 77 1F1
B9F60 6E 09 D6 09 01 6E 08 77 1E7
B9F68 BC 77 6E 8C 8C 8C 8C 8C 8C
B9F70 77 41 65 65 65 65 65 65 65
B9F78 00 96 00 96 00 96 00 96 00
B9F80 61 39 77 55 49 1B 00 05 12F
B9F88 7F 09 77 65 59 09 6A 131
B9F90 00 65 65 65 65 65 65 65 65
B9F98 65 00 05 05 05 05 05 05 05
B9FA0 00 35 00 C2 C4 C5 CB 10B

```

```

BFAB CC CD C1 42 C3 44 C5 4B 1B0
BFBO CC 53 D0 4E DA 5A A0 4E 15F
BFBB C3 43 A0 50 CF 50 C5 50 12A
BFCD A0 4D A0 41 C6 49 DB 49 1FE
BFCD D9 20 4A 2B 43 A9 45 52 155
BFDD 52 4F D2 3F 43 43 C6 2F 12D
BFDB A3 50 CC 27 44 41 C1 F3 1BF
BFEO 44 C9 FB 45 C9 D9 45 5B 1BC
BFEB DB 76 4B 41 4C D4 17 52 160
BFFO 4C C1 07 52 4C 43 C1 1F 1D5
BFBB 52 52 C1 0F 52 52 C1 1C 11C
C000 37 53 43 C6 00 FB 8B 2E 139
C008 FF CE 03 41 44 C3 FB 80 190
C010 26 FF C6 03 CF 09 0B 41 10F
C01B 44 C4 FB A0 01 FF E6 02 1B8
C020 41 4E C4 C7 C4 13 FF CD 1BD
C02B 14 43 41 4C CC FB 8B 01 161
C030 FF FE 02 43 D0 C7 05 06 1E4
C03B CF 0B 04 44 45 C3 FF 10 139
C040 17 44 4A 4E DA FF EB 18 1CF
C04B FF E3 1F FF 0B 1A 45 DB 139
C050 FF DB 1C 49 CE C7 04 06 1DE
C05B CF 03 04 49 4E C3 FF E9 11B
C060 15 C7 C2 13 FF C3 14 4A 1D1
C06B D0 E7 20 16 FF 1B 17 4A 165
C070 D2 EF 02 11 EF 0A 12 C0 19F
C07B 40 09 C7 06 0A CF 01 0B 1FB
C080 FF 2A 0C FF 22 0D FF 3A 19C
C08B 23 FF 32 24 FF F9 10 4C 1CC
C090 C4 FF D3 1E 4F 55 DA FF 12B
C09B 00 00 4E 4F D0 FB 80 01 116
C0A0 FF F6 02 4F D2 CF C1 20 1CB
C0AB 50 4F D0 CF C5 20 50 55 1CB
C0B0 53 CB FF C9 00 C7 C0 25 1BF
C0BB 52 45 DA C7 C7 1B 52 53 1B9
C0CB 04 FB 9B 26 FF DE 03 53 1BD
C0CB 42 C3 FB 90 01 FF D6 02 165
C0D0 53 55 C2 FB AB 01 FF EE 1FB
C0DB 02 5B 4F D2 00 C0 C0 40 17B
C0E0 05 42 49 DA C0 B0 05 52 1FB
C0EB 45 D5 C0 C0 05 53 45 D4 10B
C0FB F8 10 01 52 CC FB 1B 01 13B
C0FB 52 D2 FB 00 01 52 4C C3 17E

```

```

C100 FB 0B 01 52 52 C3 FB 20 1B0
C10B 01 53 4C C1 FB 2B 01 53 1D5
C110 52 C1 FB 3B 01 53 52 CC 1B5
C11B 00 A9 43 50 C4 B9 43 50 14C
C120 44 D2 A1 43 50 C9 B1 43 107
C12B 50 49 D2 AA 49 4E CA BA 12A
C130 49 4E 4A D2 A2 49 4E C9 1AF
C13B B2 49 4E 49 D2 AB 4C 44 19C
C140 C4 B8 4C 44 44 D2 A0 4C 10E
C14B 44 C9 B0 4C 44 49 D2 44 1AC
C150 4E 45 C7 B8 4F 54 4A D2 1CE
C15B B3 4F 54 49 D2 AB 4F 55 1C0
C160 54 C4 A3 4F 55 54 C9 4D 1C9
C16B 52 45 54 C9 45 52 45 54 1E4
C170 CE 6F 52 4C C4 67 52 52 1AA
C17B C4 00 CF 4A 07 41 44 C3 12C
C180 C7 46 27 49 CD C7 40 1D 16E
C18B 49 CE CF 4B 0E CF 43 0F 160
C190 F7 57 21 F7 47 22 4C C4 1DF
C19B C7 41 1F 4F 55 D4 CF 42 1B0
C1A0 07 53 42 C3 00 FF 00 00 15E
C1AB 00 00 00 B6 09 BE 19 96 1CC
C1B0 21 9E 22 A6 23 AE 29 B6 137
C1BB 34 BE 35 2B 36 E1 39 E3 1B5
C1C0 46 E5 4E E9 56 F9 5E 6E 17D
C1CB 66 71 70 73 72 75 74 7E 193
C1D0 77 2A 90 B5 00 00 01 01 1EB
C1DB 2A 2A 20 53 45 4C 46 20 1BE
C1E0 52 45 4C 4F 43 41 54 41 14B
C1EB 42 4C 45 20 44 45 42 55 113
C1F0 47 47 45 52 20 56 2E 34 1FD
C1FB 2E 30 20 2A 2A 0D 46 52 177
C200 4F 4D 3F 0D 20 54 4F 3F 1EA
C20B 0D 45 5B 45 43 55 54 45 120
C210 3F 0D 46 49 4C 45 20 4E 1DA
C21B 41 4D 45 3F 20 0D 46 52 1D7
C220 4F 4D 0D 20 54 4F 0D 4C 1C5
C22B 4F 41 44 3F 0D 20 54 4F 1E3
C230 50 3F 0D 56 45 52 49 46 11B
C23B 59 3F 0D 56 45 52 49 46 121
C240 59 49 4E 47 20 0D 4F 4B 1FE
C24B 0D 45 52 52 4F 52 0D 20 1E4
C250 44 41 54 41 3F 0D 44 45 1EF

```

```

C25B 46 4D 20 22 0D 20 41 44 1B7
C260 44 52 45 53 53 20 43 41 125
C26B 4C 43 55 4C 41 54 45 0D 117
C270 20 3D 20 24 0D 20 3D 20 12B
C27B 20 24 0D 20 3D 20 4F 56 1B8
C280 45 52 21 0D 20 20 41 44 1BA
C28B 44 52 20 43 55 4E 54 13F
C290 0D 20 41 20 20 46 20 20 134
C29B 42 20 20 43 20 20 44 20 169
C2A0 20 45 20 4B 20 20 4C 179
C2AB 0D 20 41 27 20 46 27 20 142
C2BB 42 27 20 43 27 20 44 27 17E
C2BB 20 45 27 20 4B 27 20 4C 1B7
C2CB 27 0D 20 50 43 20 20 20 147
C2CB 53 50 20 20 20 49 5B 20 1C4
C2DB 20 20 49 59 20 20 20 49 1BB
C2DB 0D 52 45 4C 4F 43 41 54 117
C2EB 45 0D 44 41 54 41 20 41 1CD
C2FB 4E 4F 20 44 41 54 41 20 1F7
C2FB 41 52 45 41 21 0D 4F 4B 1E1
C300 3F 0D 06 C7 C6 C7 DB F7 17B
C30B CB FF 01 CF 22 E7 C2 C7 12C
C310 C4 C7 C3 FF CD FF F0 46 1AF
C31B C7 70 FB B6 C7 34 FE 36 1E4
C320 FF 21 FF 2A FF 22 FF CB 134
C32B FF 43 C7 F0 10 1B 20 2B 169
C330 30 3B F0 0D 20 AE 33 AE 107
C33B 13 00 B0 4F 2F AE 33 00 122
C340 90 00 20 AF 13 09 20 90 12B
C34B 0D EB 03 7C 2B 0D D0 76 195
C350 25 FC FF 00 01 05 BB F2 1D3
C35B BA 3E 11 00 CF B0 C3 00 11B
C360 00 00 00 50 3F BF 00 00 14E
C36B 7B C3 34 1B 0B 00 02 00 191
C370 00 00 00 00 00 00 00 00 100
C37B 00 00 00 00 5D 4C 0D 1C3
C38B 0D 0D 0D 0D 0D 0D 0D 0D 16B
C39B 0D 0D 0D 0D 0D 0D 0D 0D 16B
C39B 0D 0D 0D 0D 0D 0D 0D 0D 16B
C39B 0D 0D 0D 0D 0D 4F 3F 0D 1DC
C3AB 45 5B 45 43 55 54 0D 1B 1F3

```

TRACER/80B

～マシン語をインタープリティブにトレース！～

●HUDSON SOFT 竹部隆司／中本伸一

(1)TRACER/80Bの概要

MZ-80Bの標準モニタSP-1520はMZ-80K/CのモニタSP-1002とはちがひ、マシン語モニタとしての性格を持っています。通常はこのモニタを使用し、マシン語プログラムをメモリ中へ格納し、実行するのですが、少々プログラムが大きくなり、正常に動作しない場合は、おかしい部分を発見するのに多くの時間を費やしてしまいます。

あるいは、マシン語はむずかしいと言ってマシン語プログラムをあきらめてしまう人もいるかもしれません。

MZ-80Bは順調に出荷され、ユーザーもマシン語プログラムを作ろうとしていることと思います。今回発表のZ80 TRACER/80Bは、このような方を対象にマシン語プログラムを能率よく開発できるように作られたマシン語開発ツールです。

Z80 TRACER/80Bは、通常のマシン語モニタの機能に加え、Z80シミュレータ、ディスアセンブラ、16個のブレーク・ポイントなど、およそマシン語モニタに求められる機能を内蔵し、使い易いデバッグとなっています。

Z80 TRACER/80Bは基本的に次の機能を持ちます。

- ①メモリ・ダンプ
- ②ディスアセンブラ
- ③レジスタ・ダンプ
- ④ブレーク・ポイント
- ⑤ブロック転送
- ⑥トレース
- ⑦実行
- ⑧プリンタ制御
- ⑨プログラムのセーブ、ロード

この内、トレースはTRACERの中で一番重要な機能です。これは、ユーザープログラムを常に監視し、指定されたレジスタやメモリ内容を表示し、内蔵シミュレータにより、インタープリティブにユーザープログラムを実行します。

ユーザーは表示された内容をモニタし、その動作を確認することにより、容易にバグを発見することができ、プログラム開発効率が向上します。

(2)ダンプ・リストの入力の仕方

HuBASICを持っている方は次の手順でダンプ・リストを

入力してください。

- i) HuBASIC/80BをLOAD.
- ii) MONでHu-MONITORを呼び出す。
- iii) Wコマンドでダンプ・リストを入力する。
- iv) Sコマンドで入力したプログラムをカセットへセーブします。

SB-5520を使用している方は次の手順で入力してください。

- i) SB-5520をLOAD
- ii) LIMIT40959
- iii) MONでMONITOR-1520を呼び出す。
- iv) ダンプ・リストを入力する。
- v) BASICへ戻る。
- vi) USR \$A000でTRACERを起動する。
- vii) TRACERのセーブ・コマンドOで自分自身をセーブします。

HuBASICまたはSB-5520で入力したTRACERは、IO CSつまり自分自身でキー入力、画面出力などのプログラムを内蔵しているので、モニタなどをトレースすることができ

ます。カセットにセーブしたTRACERは、モニタよりロードし、A000Hより走らせることにより、モニタの管理を離れるので、これ以降モニタが存在していなくても正常に動作します。

(3)Z80TRACER/80Bのコマンド

TRACERが起動されるとプロンプト“T:”が出力されます。このプロンプトが出ているときにコマンドが入力できます。なお、各コマンド中で使用されるアドレス、データはすべて16進数です。

●メモリ・ダンプ

機能

指定されたメモリの内容を表示します。メモリの16進ダンプのほか、ASCIIダンプも同時に表示します。また、この表示させた内容はカーソル・エディットが可能です。文法

●MADR1—ADR2 ADR1からADR2までを表示する。

●MAOR ADRから128バイトを表示する。

●M これ以前に表示されたメモリ・ダンプの次のアドレスより、128バイトを表示します。

- メモリ・チェンジ 変更したいところまでカーソルを持っていき、16進2桁を入力します。ASCII ダンプは変更できません。ただし、16進ダンプの所に";"と1文字を入力すると、そこにその文字のASCII コードが入ります。

```
8000 00 00 00 00 00 00 00 .....
           ↳カーソル・エディット
8000 ;H ;U ;D ;S ;O ;N 00 .....
           ↳8000をメモリ・ダンプ
8000 48 55 44 53 4F 4E 00.....
```

②ディスアセンブル・リスト

機能

指定されたメモリ・アドレスをプログラムの先頭とみなし、それ以降をディスアセンブルします。メモリ内容およびニック・コードを表示しますが、メモリ内容の方はカーソル・エディットできます（ニック・コードの方はカーソル・エディットできません）。

文法

- L**ADR1,ADR2 ADR1からADR2までのディスアセンブル・リストを表示します。
- L**ADR ADRから22行分のディスアセンブル・リストを表示します。
- L** これ以前に表示されたディスアセンブルした次のアドレスよりディスアセンブル・リストを表示します。

③レジスタ・ダンプ

機能

CPUのレジスタの内容を表示します。レジスタ・ダンプは、レジスタ・チェンジの機能も持ち、カーソル・エディットが可能です。

文法

- A** 表レジスタ（A, B, C, D, E, H, L）を表示します。
- C** 裏レジスタ（A', B', C', D', E', H', L'）を表示します。
- P** ポインタ関係のレジスタ（PC, SP, IX, IY）を表示します。
- F** フラグ・レジスタを表示します。
- R** 上記A, C, Pを同時に表示します。
- S** スタック・トップより8バイトを表示します。

④ブレーク・ポイント

機能

ユーザープログラム中にブレーク・ポイントを設定し、トレース中にそのポイントを通過すると、あらかじめ設定された動作を行います。これを使用するとユーザーの見た目のプログラム中のある場所の状態を、確実に把握することができます。

文法

(1)モード・エレメントm

モード・エレメントとは、ブレーク・ポイントを通過したときに実行する機能のことで、次のエレメントを持ちます。

- A, C, P, F, S** 各レジスタを表示（レジスタ・ダンプと同じ機能）
- X** 上記A, C, P, F, SをXで、まとめて代表します。
- I** ブレーク・ポイントを設定したアドレスにあるプログラムのニック・コードを表示します。
- B** TRACERのコマンド待ちに戻ります。Bを設定しなければ、ブレーク・ポイントを通過してもコマンド

待ちには戻らず、それ以外に設定されたエレメントを実行します。

- H** 表示をHOMEに指定します。

(2)ブレーク・ポイントの設定

- B**ADR, m ADRにブレーク・ポイントを設定し、モード・エレメントmを実行します。モード・エレメントは必要なものを列記し、順不同でデリミタは必要ありません。
- B**ADR1, m/ADR2, n ブレーク・ポイントはADR1に設定され、このポイントを通過するとモード・エレメントmを実行し、さらにADR2よりnバイトのメモリ内容を表示します。

(3)ブレーク・ポイントの解除

- B**ADR ADRにセットされているブレーク・ポイントを解除します。
- B**& すべてのブレーク・ポイントを解除します。
また現在どこにどのようなブレーク・ポイントが設定されているかを見る場合は、Bのみを入れてください。この表示されたブレーク・ポイントはカーソル・エディットが可能です。

なお、ブレーク・ポイントは、最大16個まで設定することが可能です。

⑤ブロック転送

機能

指定されたメモリ空間を別の空間へ転送します。自分自身に重ねて転送することも可能です。

方法

- X**ADR1,ADR2,ADR3

ADR1よりADR2までのメモリ内容を、ADR3より始まるアドレスへ転送します。

⑥トレース

機能

ユーザーのマシン語プログラムを、インタープリティブにトレースします。トレース・モードはいろいろとあり、必要に応じて選択できます。

方法

- T**ADR ADRよりトレースを開始します。
- T** ブレーク・ポイントまたは **[SHIFT] + [BREAK]** でトレースを停止したアドレスより、トレースを再開します。
- #m** トレース・モードのときにどのような動きをするかをmで指定します。mはモード・エレメントで、ブレーク・ポイントの所で説明したものと同じものです。
- #&** トレース・モードときのモード・エレメントを解除します。
- #** 現在セットされているモード・エレメントを表示します。このエレメントはカーソル・エディットが可能です。
- IT** すべてのインストラクションをトレースします。これをセットし、Tを実行するとトレースの開始です。
- IN** CALL命令以外をトレースするようにセットします。つまりサブルーチンはトレースしません。

TADRまたはTを実行する前に必ず、上記のモードを設定してください。またトレース中に **[SHIFT] + [BREAK]** を押すと、トレースを中断し、コマンド待ちに戻り、**[スペース]** を押すと、一時停止します。もう1度 **[スペース]** を押すと、1命令をトレースし、また一時停止します。他のキーを押すと連続的にトレースを再開します。

②实行

機能

トレーサの管理を離れ、直接ユーザーのマシン語プログラムを実行します。

文法

- **GADR** ADRより始まるプログラムヘジャンプし、トレサの管理を離れます。

④ プリンタ制御

機能

トレース内をプリンタへ出力します。ただし、同一
 窓がCRTにも出力されます。もしプリンタが接続されてい
 なければ、メッセージを出力し、プリンタ・モードを解除
 します。プリンタ命令はスイッチになっていて、再度入
 するとプリンタ・モードが解除されます。

文法

■トレーサ ダンプ・リスト■

[illegible]

●& プリント・モードのセットと解除を行ないます。

⑨ユーザープログラムのセー

現在メモリ中にあるプログラムをカセットへ出力します。

74

- **O**ADR1 ADR2 ADR3 **ファイルネーム**
ADR 1 はスタート・アドレス、ADR 2 はエンド・アド

レス、ADR 3 はプログラマ

1

機能

- Y: ファイル名は付けませんが、カセットの最初のファ



MZ-80B

BASIC SB-5520 の拡張



■山川孝之

最近のBASICの機能の充実ぶりには目を見張るものがあり、使い切れないほどのコマンド、ステートメントを持った機種がぞくぞくと登場しています。そういう状況の中で、我が所有するSHARP BASIC は比較的命令数の少ない機種に属しており、少々不満に感じていました。

そこでRAM BASICの特長を生かし、機能の強化を行なうことにしました。強化するステートメント、コマンドは、他機種から見ればありきたりのものばかりですが、中には従来のBASICと趣きを異にした機能も付加しています。

拡張された機能一覧表

機 能	内 容
サーチ機能	BASICテキストの文字列を捜す。
ラベル・ジャンプ機能	GOTO, GOSUB, RUNなどのオペランドとしてラベルが使用できる。
段 付 け	リストを見やすくするために段付けができる。
IF~THEN~ELSE~IEND	ELSE文を追加し、さらにIENDでIF文の範囲を指定することにより、THENおよびELSE以降を複数行に渡って記述できるようにした。
REPEAT~UNTIL	条件が成立するまでREPEATとUNTIL間を繰り返すステートメント。
WHILE~WEND	条件が成立している間、WHILEとWEND間を繰り返すステートメント。
USING*#ss.ss#	数値出力のフォーマット指定を行なう。
VPOKE	ディスプレイに直接データ(キャラクター)を書く。
VPEEK	ディスプレイのデータ(キャラクター)を読み取る。
DELETE	テキストの任意の行を消す。
VLIST	数値変数、ストリング変数の名前とその内容を表示する。

各コマンド、ステートメントの使い方

＜サーチ機能＞

この機能はI/O別冊のシステム・プログラム・ライブラリ②で発表したMZ-80K用(SP-5010)のサーチ機能を80B用に移植したものですから詳しくはそちらを参照してください。

以下使い方のみ述べます。

① S. [CR]

サーチの初期設定。サーチを始める前に必ず実行します。

② S. 文字列 [CR]

ダブルクォーテーション・マーク(“)で囲まれた以外の文字列をサーチします。文字列としては英、数字、英記号の組み合わせです。グラフィック文字やカナを入力すると“ERROR 1”を表示します。

③ S“文字列 [CR]

ダブルクォーテーション・マーク(“)で囲まれた文字列をサーチします。文字列としてはすべてのキャラクタを指定できます。

④ S/ [CR]

上記②または③の文字列を連続してサーチするときに使います。これにより同一の文字列を連続して捜すのが楽になります。

＜ラベル・ジャンプ機能＞

GOTO, GOSUB, RUN, ON~GOTO, ON~GOSUBのオペランドとしてラベルを指定することができます。

ラベルの書式

- 1) ラベルは必ずダブルクォーテーションでくくってください。
- 2) ラベルはすべてのキャラクタが使用可。
- 3) ラベルの長さは1行内に納まるならば制限はありません。
- 4) ラベルは必ず行の先頭に書いてください。行の途中に書いてもエラーにはなりませんが、ラベルとして認識しません。もちろんラベルとして認識させるのは簡単ですが、ラベルのサーチ・スピードを速くするためにこのような制約を付けました。

インタプリタの場合、特にそうだと思いますが、一般に文法上の制約を強くするほど自由度はなくなりますが、スピードは向上するようです。

＜段付け＞

SB-5520では段付けをしようとしても、インタプリタの方で親切にもスペースを詰めてくれましたが、この親切をとり除きました。これにより見やすいプログラムを作ることができます。

＜IF~THEN~ELSE~IEND＞

ELSE文はよく存じのように、IF文で条件式が成立しないときにTHEN以降の文を飛ばしELSE文以降を実行するためのステートメントで、すでにこの機能を持つ機種は多くありますが、今回の機能追加ではさらに一歩進め、IENDステートメントでIF文の範囲を指定することによりTHENおよびELSE以降を複数行に渡って記述できるようにしました。これによりGOTO文の使用を大幅に減らすことができ、前述の段付け機能と相まって見やすいプログラムを記述することが可能です。

① 文法

I/O'82年3月号記載のKコンパイラのIF文とはほぼ同じと考えて差し支えありません。すなわち、IF文のネストには制限ありませんが、1つのIF文に対しては必ず対応するIEND文が必要で、対応するIEND文がないときはERROR番号35と最終の文番号を表示します。

② 使用例 (SB-5520との対応)

SB-5520	移植されたBASIC(SB-5522と命名)
10 IF A=1 GOTO 30	10 IF A=1 GOTO 30 IEND
10 IF A=1 THEN B=C GOTO 40	10 IF A=1
20 B=D	20 THEN B=C
30 ...	30 ELSE B=D
	40 IEND
	1行に書いても可
10 IF A=1 GOTO 50	10 IF A=1 THEN
20 IF B=1 THEN X=10 GOTO 40	20 IF B=1
30 X=20	30 THEN X=10
40 Y=10 GOTO 70	40 ELSE X=20
50 IF C=1 THEN X=30 GOTO 70	50 IEND
60 X=40	60 Y=10
70 Z=100	70 ELSE
	80 IF C=1
	90 THEN X=30
	100 ELSE X=40
	110 IEND
	120 IEND
	1行に書いても可、ただし、50行と60行の間は*:*が必要。(IEND: Y=10:)
10 IF A=1 THEN X=20 GOTO 30	10 IF A=1 THEN
20 IF C=1 THEN X=10	20 IF C=1 THEN X=10 IEND
30 ...	30 ELSE X=20 IEND

<REPEAT~UNTIL>

UNTILの次に書かれた条件式が成立するまで、REPEATとUNTILの間のステートメントを繰り返します。GOTOでループを組むよりも文番号サーチがないだけスピードが速くなり、段付け機能と相まってプログラムが見やすくなります。

① 文法

1) REPEATはオペランドを持たないステートメントですから、マルチステートメントにする場合は必ずセパレータ";"を入れてください。

(誤)

(正)

10 REPEAT A=A+1→10REPEAT: A=A+1

UNTILはIF文と同じ性格のステートメントですからUNTILの次には必ず条件式が必要です。

2) ネスティング・レベル15段です。それ以上ネストしようとするときERROR番号11を表示します(FOR~NEXTが16を越えた場合のエラー番号と同じ)。また、対応するREPEAT文がないのにUNTIL文が使われた場合ERROR番号13を表示します(FOR文がないのにNEXTが使われた場合のエラー番号と同じ)。

② 使用例

[Y]キーが押されるまで待つプログラム

10 REPEAT: GET A\$: UNTIL A\$="Y"

<WHILE~WEND>

WHILEの次に書かれた条件式が成立している間、WHILEとWENDの間のステートメントを繰り返します。

① 文法

- 1) WHILEはIF文と同じ性格のステートメントなので、WHILEの次には必ず条件式が必要です。
- 2) ネスティング・レベルは15段です。それ以上ネストしようとするときERROR番号11を表示します。また、対応するWHILE文がないのにWENDが使われた場合はERROR番号13を表示します。

<USING>

USING命令は数値の表示のときその表示フォーマットを指定する命令です。

① 文法

必ずPRINT文と対で使います。基本形として次の5種があります。

- 1) PRINT USING...フォーマット指定を解除する。
- 2) PRINT USING "####..."符号を含め整数4桁を表示。
- 3) PRINT USING "###..."符号を含め整数3桁と小数点を表示。
- 4) PRINT USING "###.###..."符号を含め整数3桁と小数点および小数点以下2桁を表示。
- 5) PRINT USING "##.###..."符号を含め整数2桁と小数点および小数点以下3桁の指数表示を行なう。

以上に示すようにUSINGの機能としては最小のもので、フォーマット指定が単独にできるので、同じフォーマットで続けて表示したい場合、一度だけフォーマット指定すれば良いようになっています。

② エラーについて

ERROR 20...このエラーはフォーマット指定時に、上記1)~5)の書式に合わない場合、または"#"の数が27を越えた場合に起こります。

ERROR 21...このエラーはフォーマット指定に合わせて数値を表示するときに、そのフォーマットでは表示しきれない場合(たとえばPRINT USING "##.###"; AでAの値が3桁以上の場合)に起こります。

③ 使用例

1) 10 PRINT USING "#####.###"
20 A=1.234: B=-234.5: C=1503
30 PRINT A: PRINT B: PRINT C: PRINT 15.999
実行結果

----1.23

---234.50

---1503.00

--- 15.99

(注) ---はスペースを示します

2) 10 PRINT USING "#####.###": A: USING "#####.###"
B

実行結果 (A=15, B=-1375.8743のとき)

---15.--- -13.758 E+02

<VPOKE>

MZ 80BのVRAMはメイン・メモリと独立しているため、MZ-80K/CのようにPOKE文でVRAMに直接キャラクタを置くことができません。このVPOKE文はこれらを可能に

するため新しく追加したステートメントです。

① 文法

書式はPOKE文と同じです。

② 使用例

VPOKE \$D000, \$41……ディスプレイのホーム位置にAを表示する(\$41はAのASCIIコード)。

<VPEEK>

先に示したのと同じ理由により、PEEKではVRAMの内容を知ることができませんが、このVPEEKでそれを可能にしました。

SB-5520ではCHARACTER\$という関数があり、VPEEK相当の機能があるためVPEEKは省略しようと思いましたが、CHARACTER\$で情報を得ることができるのはキャラクタ用VRAMの内容だけで、グラフィック用GRAMのバイト情報は得ることができないということで、このVPEEKを付加することにしました。またMZ-80K/C用のソフトをMZ-80Bに移植する際にもこのVPEEKがあった方が移植し易いと思います(スピードもVPEEKの方が2~3割速い)。

① 文法

書式はPEEK文と同じです。

② 使用例

A=VPEEK(\$D000)……ディスプレイのホーム位置にあるキャラクタのASCIIコードを読み、Aに代入する。

<DELETE>

① 使い方

- 1) DELETE n 行番号nを削除する。nがないときは何もなし。
- 2) DELETE - n 最も若い行番号からnまでを削除する。
- 3) DELETE n-m 行番号nからmまでを削除する。
- 4) DELETE n- 行番号n以降すべてを削除する。

<VLIST>

数値変数およびストリング変数で、すでに登録されたものについてその変数名と内容を表示します。ただし、配列変数については表示しません(サーチ・コマンドでDIM文を捜せば、使った配列変数の名前を知るができます)。

① 使い方

- 1) VLIST N……登録された数値変数の名前とその内容を表示する。
- 2) VLIST S……登録されたストリング変数の名前とその内容を表示する。
- 3) VLIST……登録された数値変数とストリング変数の名前とその内容を表示する。

(注)表示するのは登録された変数のみですから、ソース・テキスト中に書かれた変数があってもその行が実行されていなければ変数は登録されないで表示はしません。

なお、変数リストを表示中スペース・キーを押すと一時ストップします。離せば表示を続行します。

<PATTERN文について>

SB-5520のPATTERN文の積重ね段数は24段までとなっています。大きな絵を書きたいときは不便なので、今回の改造で200段まで可能としました。と言うと大げさに聞こえますが、44DF番地の19H(25₁₀)をC9H(201₁₀)に変更しただけです。何のために24段までに制限されたのかはわかりませんが、いまのところ正常に動作しています。しかしストリングの長さの最大255ですから、むやみに大きな絵を一度に描くことはできません。

SB-5520の拡張の仕方 拡張BASICの作り方

- ① BASICの拡張プログラムをORG \$5160でアセンブルする(アセンブラのない人は左側のマシン・コードのみをモニタを使って\$5160から入力する)。
 - ② できたマシン語プログラムを\$7000へブロック転送し、テープにセーブする。
 - ③ BASIC書き換えプログラムをORG \$7B30でアセンブルする(アセンブラのない人は\$7B30よりマシン・コードのみを入力)。
 - ④ ②で作ったテープを読み込み、2つまとめて\$7000~\$7C7Fまでをオート・スタート\$00B1としてセーブする。
 - ⑤ SB-5520をIPLよりロードする。
 - ⑥ MON[CR]でSB-1520へ移る。
 - ⑦ ④で作ったテープを読み込む。
 - ⑧ 新しいテープをセットしてJ- \$7B30を行なう。
- 以上の操作により拡張されたBASICテープができる。

プログラムについて

1つ1つの機能について、そのプログラムの内容を述べることはとてもできませんので、構成のみ述べます。

ステートメント、コマンドを拡張しようと考えたとき、以前の80K/CのBASICのように空きエリアがないのかと捜しましたが、それらしき所はあったものの確信が持てず、また、大ききも1Kバイト以下のため空きエリアはあきめ、テキスト・エリアを高位番地へ押し下げたことにより拡張用のエリアを確保しました。これにより従来のテキスト格納番地が\$515Cだったのが、今回の変更により\$600C番地になりました。したがってフリーエリアも44268バイトから40520バイトに減少しました。

次に拡張用のステートメント、コマンドをどのように中間言語に変換しようかと悩みましたが、よく調べてみると拡張用に準備された部分を見つけることができました。つまり、1531番地および15B D番地の2バイトを拡張コマンドを書いたアドレス(今では5299番地)に書き直せば良いのです。このとき、中間言語は\$81+ \$〇〇となります。次に、このように作られた中間言語の実行番地を求めるプログラムが必要となりますが、このためには1C 63番地の2バイトをコントロール・プログラムの書かれたアドレス(今では5300番地)に変更すれば良いことになります。コントロール・プログラムは次のようになります。

5300	21	DA	53	LD	HL, EXECAD	実行アドレステーブル先頭番地
5303	3D			DEC	A	中間言語\$81+〇〇?7
5304	C2	33	14	JP	NZ, \$1433	ERROR 1表示(NOのとき)
5307	C3	65	1C	JP	\$1C65	その実行アドレスを求める (YESのとき)

あとはその実行番地から拡張ステートメント、コマンドを処理するプログラムを書いていきました。

アセンブル・リストについて

- ① 今回発表のアセンブル・リストはSHARPアセンブラと一部モニタが異なっているので以下にその対照を示します。

今回発表のアセンブラ	SHARPアセンブラ
DB	DEFB
DW	DEFW
DS	DEFS
DM	DEFM

注) 16進表示は○○Hではなく\$○○となっている。

② ラベルでFR××××(××××は16進4桁)というのがありますが、これはインプットの××××番地からこのラベルの所へジャンプしてすることを示しています。

③ 今回発表のプログラムでSB-5520内のルーチンをコールしている部分がありますが、自分でわかっている範囲でその機能を述べます。

アドレス	内 容	備 考
13B2	テキストの修正、インサートによりテキスト・エンド・アドレスや変数テーブル・アドレスを変更する。	
14E1	(\$14EB)=0	
14E6	(\$5159)=0	
14F2	入力されたASCIIコードを中間変数に\$4B10よりストアする	
180A	(HL)がblank・コードでなくなるまでHLをインクリメントする。	
181A	*で囲まれた部分を無視しHLをインクリメントする。	
182D	BC-BC+1(BCの補数に1を加えたものをBCとする)	
1835	(HL)が0-9の数字かどうか、YESのときキャリセット。	
183E	(HL-DE) HLとDEを比較する。	
1854	HL-(\$F94H)+5	

1860	CALL \$1860	このような使い方を。すなわち(HL)の内容が\$81かどうか調べYESならHLを+1し、次の2バイトを飛ばしてCP \$80を実行、NOなら?IFのラベルへジャンプしHLは+1しない。
18C1	HLで示されるHLで示されるアドレスからの10進ASCIIコードを2進にしてDEへ。	
197B	HLで示されるHLで示される文番号をサーチし見つければZフラグをセットし、その文番号のあるアドレスをHLへ。	
19F1	式がストリングかどうか調べYESならエラー。	
1A06	(HL)が*?または0D(改行)かどうかチェック (すなわちステータメントの終わりがどうか)、YESならZフラグをセット。	
1A84	(HL)=0, (HL+1)=0	
1DC2	LIST n-mのn, mを2進にしてそれぞれ(\$4B83), (\$4B85)にストアする。	
2365	乗法式を解析し、その結果(0または-1)を浮動小数点にして\$4F94以降5バイトに書く。	
27E3	(HL)が*?かどうか、NOならERROR1。	
2A48	PEEK(××××)やPOKE(××××)の××××を2進にしてDEへ。	
3118	HLで示されるアドレスから5バイトに格納されている浮動小数点バイナリを10進ASCIIコードに変換し、DEで示されるアドレスより書き、最後に改行マーク\$0Dを書く。	
40EB	(HL)が*?であればそれを無視し、HLをインクリメントする。そうでなければ何もしない。	

BASIC拡張プログラム・リスト

5160	ORG \$5160	510B D1	LO3:	POP	DE
5160	I	510C 13		INC	DE
5160	I	510D 13		INC	DE
5160 4F	FR13601 YOR A	510E 13		INC	DE
5161 324452	LD (STRFLG), A	510F 13		INC	DE
5164 EB	EX DE, HL	5110 1A	LO4:	LD A, (DE)	
5165 C00A10	CALL \$180A	51E1 FE22		CP \$22	
5168 EB	EX DE, HL	51E3 201C		JR NZ, L05	
5169 1A	LD A, (DE)	51E5 3A4452		LD A, (STRFLG)	
516A FE33	CP \$F	51E8 B7		OR A	
516C C2F214	JP NZ, \$14F2	51E9 2019		JR NZ, L06	
516F 13	INC DE	51EB 13	LO9:	INC DE	
5170 1A	LD A, (DE)	51EC 1A		LD A, (DE)	
5171 1B	DEC DE	51ED FE0D		CP \$0D	
5172 FE22	CP \$22	51EF 2019		JR Z, L0B	
5174 201E	JR Z, STRSER	51F1 FE22		CP \$22	
5176 FE2E	CP \$F	51F3 20F6		JR NZ, L09	
5178 201F	JR Z, STMSER	51F5 13		INC DE	
517A FE2F	CP \$F	51F6 1A		LD A, (DE)	
517C C2F214	JP NZ, \$14F2	51F7 13		INC DE	
517F 015000	LD BC, \$00	51F8 FE0D		CP \$0D	
5182 110E4B	LD DE, \$4B0E	51FA 2B05		JR Z, L01	
5185 214952	LD HL, SERBUF	51FC 1B		DEC DE	
5188 EDB0	LDIR	51FD 18E1		JR L04	
518A 7A134B	LD A, (\$4B13)	51FF 13		INC DE	
518D FE22	CP \$22	5200 1A		LD A, (DE)	
518F 200B	JR NZ, SERCH	5201 BE	LO5:	CP (HL)	
5191 C33C52	JP STRFT	5202 2B09		JR Z, 7C0INC	
5194 3E01	STRSER: LD A, 1	5204 13	LO6:	INC DE	
5196 324452	LD (STRFLG), A	5205 1A		LD A, (DE)	
5199 C0F214	STMSER: CALL \$14F2	5206 FE0D	LO7:	CP \$0D	
519C 2100A0	SERCH: LD HL, \$6000	5208 20D6		JR NZ, L04	
519F 3A144B	LD A, (\$4B14)	520A 13	LO8:	INC DE	
51A2 FE0D	CP \$0D	520B 18B4		JR L01	
51A4 200C	JR NZ, SRCHGO	520D 13	7C0INC:	INC DE	
51A6 224552	LD (SERADR), HL	520E 23		INC HL	
51A9 210000	LD HL, 0	520F 7E		LD A, (HL)	
51AC 224752	LD (SERPNT), HL	5210 FE0D		CP \$0D	
51AF C30A13	JP \$130A	5212 2B09		JR Z, 7C0INC	
51B2 EB	SRCHGO: EX DE, HL	5214 1A		LD A, (DE)	
51B3 2A4752	LD HL, (SERPNT)	5215 BE		CP (HL)	
51B6 7C	LD A, H	5216 26F5		JR Z, 7C0INC	
51B7 B5	OR L	5218 21144B		LD HL, \$4B14	
51B8 2B04	JR Z, L00	521B 18E9		JR L07	
51BA ED54552	LD DE, (SERADR)	521D 21834B	COINC1:	LD HL, \$4BB3	
51BE 21144B	LO0: LD HL, \$4B14	5220 CDB41A		CALL \$1A84	
51C1 D5	LO1: PUSH DE	5223 2A4752		LD HL, (SERPNT)	
51C2 D9	EXX	5226 22B34B		LD HL, (\$4BB3), HL	
51C3 E1	POP HL	5229 22B54B		LD HL, (\$4BB5), HL	
51C4 114552	LD DE, SERADR	522C D9		EXX	
51C7 010400	BC, 4	522D 015000		LD BC, \$0	
51CA EDB0	LDIR	5230 114952		LD DE, SERBUF	
51CC D9	EXX	5233 210E4B		LD HL, \$4B0E	
51CD D5	PUSH DE	5236 EDB0		LDIR	
51CE 0404	LD B, 4	5238 D9		EXX	
51D0 13	INC DE	5239 C3561D		JP \$1D56	
51D1 1A	LD A, (DE)	523C 3E01	STRFT:	LD (STRFLG), A	
51D2 B7	OR A	523E 324452		JP SERCH	
51D3 2006	JR NZ, L03	5241 C39C51			
51D5 10F9	DJNZ L02	5244 00	STRFLG:	DB 0	
51D7 D1	POP DE	5245 0000	SERADR:	DW 0	
51D8 C3F612	JP \$12F6	5247 0000	SERPNT:	DW 0	
		5249	SERBUF:	DS \$0	

5299 454C53C5	CHDTBL	DB	"E","L","S",#C5	538E FE0D	CP	#0D
529D 52455045		DB	"R","E","P","E","R",#D4	5390 C2561C	JP	NZ,#1C56
52A1 41D4				5393 C30C1C	JP	NZC
52A3 554E5449		DB	"U","N","T","I",#CC	5396 23	FIEND	INC HL
52A7 CC				5397 3A5A54	LD	A,(IECONT)
52AB 574B494C		DB	"H","H","I","L",#C5	539A B7	OR	A
52AC C5				539B CAF353	JP	Z,IEND
52AD 55454EC4		DB	"W","E","N",#C4	539E 3D	DEC	A
52B1 49454EC4		DB	"I","E","N",#C4	539F 3A5A54	LD	(IECONT),A
52B5 5553494E		DB	"U","S","I","H",#C7	53A2 C35E53	JP	ELBE
52B9 C7				53A5 CD601B	CALL	#1B60
52BA 4552524F		DB	"E","R","O","R",#D2	53AB 0D	DB	#0D
52BE D2		DB	"E","R",#D2	53A9 B522	DW	#2B5
52BF 4552D2		DB	"E","R",#D2	53AB CDF653	CALL	LDIRLN
52C2 4552CE		DB	"E","R",#CE	53AE CDF653	CALL	ENDCHK
52C5 52455355		DB	"R","E","S","U","H",#C5	53B1 C2B522	JP	NZ,#2B5
52C9 4D5C				53BA 3E24	LD	A,36
52CB 4D455247		DB	"H","E","R","O",#C5	53BA C3B914	JP	#14B8
52CF C5				53B9		
52D0 43495243		DB	"C","I","R","C","L",#C5	53B9	ELSE1	
52DA 4CC5				53B9		
52DB 44454C45		DB	"D","E","L","E","T",#C5	53B9 AF	ELSE1	XOR A
52DA 54C5				53BA 325A54	LD	(IECONT),A
52DC 505553C8		DB	"P","U","S",#C8	53BD CDOC54	SIEND	CALL INC"
52E0 504FD0		DB	"P","O",#D0	53CD CD601B	CALL	#1B60
52E3 564C4953		DB	"V","L","I","S",#D4	53C3 80	DB	#80
52E7 D4				53CA CD53	DW	?IEND1
52EB 56504F4B		DB	"V","P","O","K",#C5	53C6 FEBB	CP	#8B
52EC C5				53CB 23	INC	HL
52ED 56504545		DB	"V","P","E","E","K",#AB	53C9 2B1E	JR	Z,IFI
52F1 4B4B				53CB 1BFO	JR	SIEND
52F3		DS	13	53CD CD601B	?IEND1	CALL #1B60
5300 210A53	EXEAD	LD	HL,EXETBL	53D0 B1	DB	#B1
5303 3D	DEC	A		53D1 E453	DW	?C1
5304 C23314		JP	NZ,#1433	53D3 FEB5	CP	#B5
5307 C351C1		JP	#1CA2	53D5 23	INC	HL
530A B953	EXETBL	DW	ELBE1	53D6 20E5	JR	NZ,SIEND
530C 6C54		DW	REPEAT	53DB 3A5A54	LD	A,(IECONT)
530E 8F54		DW	UNTIL	53DB B7	OR	A
5310 D757		DW	WHILE	53DB C15	JR	Z,IEND
5312 B350		DW	WEND	53DE 3D	DEC	A
5314 F353		DW	IEND	53DF 325A54	LD	(IECONT),A
5316 3314		DW	#1433	53E2 1B09	JR	SIEND
5318 3314		DW	#1433	53E4 CD2354	?C1	CALL INCOD
531A 3314		DW	#1433	53E7 1B04	JR	SIEND
531C 3314		DW	#1433	53E9 3A5A54	FIFF	LD A,(IECONT)
531E 3314		DW	#1433	53EC 3C	INC	A
5320 3314		DW	#1433	53ED 325A54	LD	(IECONT),A
5322 3314		DW	#1433	53F0 1B0C	JR	SIEND
5324 2357		DW	DELETE	53F2 23	INC	HL
5326 3314		DW	#1433	53F3 C3FF1B	IEND	JP #1BFF
5328 3314		DW	#1433	53F6		
532A 5454		DW	VL1ST	53F6 11A051	LDIRLN	LD DE,#514D
532C 2156		DW	VPOKE	53F9 010400	LD	BC,4
532E 0756		DW	VPEEK	53FC ED80	LDIR	
5330				53FE C9	RET	
5330				53FF		
5330				53FF E5	ENDCHK	PUSH HL
5330				5400 2A4D51	LD	HL,#514D
5330 DDCB047E	FR22AF	BIT	7,(IX+4)	5403 7C	LD	A,H
5334 C2A553		JP	NZ,?THEN	5404 B5	OR	A
5337 AF		XOR	A	5405 2B03	JR	Z,FIERR
5338 325954		LD	(IFCONT),A	5407 7E	LD	A,(HL)
533B 325A54		LD	(IECONT),A	540B 23	INC	HL
533C CDOC54	ELSE1	CALL	INC"	5409 B4	OR	(HL)
5341 CD601B	SERELS	CALL	#1B60	540A E1	FIERR	POP HL
5344 B1		DB	#B1	540B C9	RET	
5345 5253		DW	?IF	540C CDOA1B	INC"	CALL #1B0A
5347 FEB0		CP	#B0	540F FE22	CP	#22
5349 2B19		JR	Z,?ELSE	5411 C0	RET	NZ
534B FEB5		CP	#B5	5412 23	INC	HL
534D 2B47		JR	Z,FIEND	5413 7E	INC	HL
534F 23		INC	HL	5414 FE22	INC	HL
5350 1BEC		JR	ELBE	5416 23	CP	#22
5352 CD601B	?IFI	CALL	#1B60	5417 2BFB	JR	Z,INC"
5355 B0		DB	#B0	5419 FE0D	CP	#0D
5356 5F53		DW	?CR	541B 20F4	JR	NZ,INC
5358 FEB8		CP	#B8	541D CDF653	CALL	LDIRLN
535A 2B19		JR	Z,FIFF	5420 C30C54	JP	INC"
535C 23		INC	HL	5423		
535D 1BDF		JR	ELBE	5423 FE0D	INC	HL
535F CD2354	?C1	CALL	INCOD	5425 23	INC	HL
5362 1B0A		JR	ELBE	5426 C0	RET	NZ
5364 23	?FELSE1	INC	HL	5427 CDF653	CALL	ENDCHK
5365 3A5954		LD	A,(IFCONT)	542A 2005	JR	NZ,NLINE
536B B7		OR	A	542C 3E23	LD	A,35
5369 2B18		JR	Z,FELSE	542E C3B914	JP	#14B8
536B 3D		DEC	A	5431 CDF653	NLINE	CALL LDIRLN
536C 325954		LD	(IFCONT),A	5434 C9	RET	
536F 3A5A54		LD	A,(IECONT)	5435		
5372 B7		OR	A	5436 CD601B	FR1BFF	CALL #1B60
5373 2B0E		JR	Z,FELSE	5439 B1	DB	#B1
5375 1B07		JR	ELBE	543A 021C	DW	#1C02
5377 23	FIFF	INC	HL	543C FEB0	CP	#B0
537B E5		PUSH	HL	543E 23	INC	HL
5379 215954		LD	HL,IFCONT	543F CAB953	JR	Z,ELSE1
537C 34		INC	(HL)	5442 FEB5	CP	#B5
537D 23		INC	HL	5444 2B4D	JR	Z,IEND
537E 34		INC	(HL)	5446 C3021C	CALL	#1C02
537F E1		POP	HL	5449 CD601B	FR1A06	CALL #1B60
5380 C35E53		JP	ELBE	544C B1	DB	#B1
5383 CD351B	FELBE1	CALL	#1B35	544D 091A	DW	#1A09
5386 DA741F		JP	C,#1F74	544F 2B	DEC	HL
5389 FE22		CP	#22	5450 FEB0	CP	#B0
538B CA4055		JP	Z,SOLBL			

BASIC拡張プログラム・リスト

```

5452 CB      RET Z
5453 FE85     CP #85
5455 CB      RET Z
5456 C3091A   JP #1A09
5459 00      IFCONT: DB 0
545A 00      IECONT: DB 0
545B         I
545B         I REPEAT INITIAL
545B         I
545B         I FR12F6: LD HL, NCOUNT+99
545B         LD HL, (NSPINT), HL
5461 AF      XOR A
5462 32CF54   LD (NSPINT), A
5465 ED70A4A LD SP, (#1A0A)
5469 C3FA12   JP #12FA
546C         I
546C         I REPEAT PROCESS
546C         I
546C         I REPEAT: CALL #180A
546F 225151   LD (#5151), HL
5472 21CF54   LD HL, NCOUNT
5475 7E      CP #10
5476 FE10     JP NC, L16ERR
547B 34      INC (HL)
547C 215251   LD HL, #5152
547F ED58CD5 LD DE, (NSPINT)
5483 010600   LD BC, 6
5484 ED8B     LD DR, DE, HL
5488 EB      EX DE, HL
5489 22CD54   LD (NSPINT), HL
548C C3FC1B   JP #18FC
548F         I
548F         I UNTIL PROCESS
548F         I
548F         I UNTIL: LD A, (NCOUNT)
5492 B7      OR A
5493 CA5414   JP Z, #1454
5496 D6A523   CALL #2365
5499 CDF119   CALL #19F1
549C D629A4F LD IX, (#4F94)
54A0 D0C0A7E BIT 7, (IX+4)
54A4 CAB854   JP Z, CONT
54A7 225151   LD (#5151), HL
54AA 2ACD54   LD HL, (NSPINT)
54AD 010600   LD BC, 6
54B0 09      ADD HL, BC
54B1 22CD54   LD (NSPINT), HL
54B4 21CF54   LD HL, NCOUNT
54B7 35      DEC (HL)
54B8 C3FC1B   JP #18FC
54BB 2ACD54   LD HL, (NSPINT)
54BE 114D51   LD DE, #514D
54C1 010600   LD BC, 6
54C4 23      INC HL
54C5 ED8B     LD DR, HL
54C7 C3FC1B   JP #18FC
54CA C34E14   L16ERR: JP #144E
54CD 0000     NSPINT: DW 0
54CF 00      NCOUNT: DB 99
54D0         DB 99
5533         I
5533         I LABEL SKIP
5533         I
5533         I FR1C56: CP #22
5535 CA151D   JP Z, #1D15
5538 DAB0     SUB #80
553A DA391E   JP C, #1E39
553D C35B1C   JP #1C5B
5540         I
5540         I LBL GOTO
5540         I
5540         I GOLBL: CALL #180A
5543 FE22     CP #22
5545 2806     JR Z, LBLGOTO
5547 CDC118   CALL #18C1
554A C3771F   JP #1F77
554D CDC355   LBLGOTO: CALL LBSRCH
5550 C31B1C   JP #1C1B
5553         I
5553         I LBL THEN
5553         I
5553         I FR22CB: CALL #1835
5556 DB      RET C
5557 C1      POP BC
5558 FE22     CP #22
555A CA4055   JP Z, GOLBL
555D FE0D     CP #0D
555F C2561C   JP NZ, #1C56
5562 C30C1C   JP #1C0C
5565         I
5565         I LBL GOSUB
5565         I
5565         I FR1F9C: CALL #180A
5568 FE22     CP #22
556A 2806     JR Z, LBLGOTO
556C CDC118   CALL #18C1
556F C39F1F   JP #1F9F
5572 CDC355   LBLGOTO: CALL LBSRCH
5575 13      INC DE
5576 ED535151 SUB: LD (#5151), DE
557A C3A51F   JP #1FA5
557D         I
557D         I LBL ON XX GOTO, GOSUB

```

```

557D         I
557D         I FR2144: CALL #180A
5580 FE22     CP #22
5582 2806     JR Z, LBLON
5584 CDC118   CALL #18C1
5587 C34721   JP #2147
558A CDC355   LBLON: CALL LBSRCH
558D 13      INC DE
558E EB      EX DE, HL
558F CD1A18   CALL #181A
5592 EB      EX DE, HL
5593 0B      EX AF, AF'
5594 B7      OR A
5595 CA1B1C   JP Z, #1C1B
5598 180C     JR SUB
559A         I
559A         I 1ST. NO. HI-SPEED SEARCH
559A         I
559A         I FR197B: LD (#198A), HL
559D EB      EX DE, HL
559E 2A4F51   LD HL, (#514F)
55A1 B7      OR A
55A2 ED52     SBC HL, DE
55A4 3806     JR C, OLONO.
55A6 210060   DRCTGO: LD HL, #6000
55A9 C3B119   STNSRC: JP #19B1
55AC 2A4D51   OLONO: LD HL, (#514D)
55AF 7C      LD A, H
55B0 B5      OR L
55B1 28F3     JR Z, DRCTGO
55B3 18F4     JR STNSRC
55B5         I
55B5         I LBL RUN
55B5         I
55B5         I FR1F6F: CALL #1835
55B8 DA741F   JP C, #1F74
55BB FE22     CP #22
55BD C27D1F   JP NZ, #1F7D
55C0 C34D55   JP LBLGTO
55C3         I
55C3         I LABEL SEARCH SUB
55C3         I
55C3         I LBSRCH: LD (LBLADR), HL
55C6 210060   LD HL, #6000
55C9 E5      PUSH HL
55CA 5E      LD E, (HL)
55CB 23      INC HL
55CC 56      LD D, (HL)
55CD 7A      LD A, D
55CE B3      OR E
55CF CA5D14   JP Z, #145D
55D2 D5      PUSH DE
55D3 23      INC HL
55D4 23      INC HL
55D5 23      INC HL
55D6 CD0A18   CALL #180A
55D9 FE22     CP #22
55DB B1      POP DE
55DC C1      POP BC
55DD EB      EX DE, HL
55DE 20E9     JR NZ, NEXSRH
55E0 EB      EX DE, HL
55E1 D5      PUSH DE
55E2 C5      PUSH BC
55E3 ED5B0556 LI: DE, (LBLADR)
55E7 23      COMPLB: INC HL
55E8 13      INC DE
55E9 1A      LD A, (DE)
55EA 8E      CP (HL)
55EB 2804     JR Z, 7FND
55ED B1      POP DE
55EE E1      POP HL
55EF 180B     JR NEXSRH
55F1 FE22     CP #22
55F3 280D     JR Z, 7FNDLBL
55F5 FE0D     CP #0D
55F7 CA3314   JP Z, #1433
55FA FE3A     CP #3A
55FC CA3314   JP Z, #1433
55FF C3E755   JP COMPLB
5602 E1      FNDLBL: POP HL
5603 C1      POP BC
5604 C9      RET
5605 0000     LBLADR: DW 0
5607         I
5607         I VPEEK
5607         I
5607         I VPEEK: CALL #2A4B
5607         I
5607         I CALL #27E3
5608 CD527E   IN A, (#EB)
560B DDEB     SET 7, A
560F CBF7     OUT (#EB), A
5611 D3E9     LD A, (DE)
5614 5F      LD E, A
5615 1600     LD D, 0
5617 DBE8     IN A, (#EB)
5619 CBF7     RES 7, A
561B D3E8     OUT (#EB), A
561D E5      PUSH HL
561E C39828   JP #2898
5621         I
5621         I VPOKE
5621         I
5621         I VPOKE: CALL #2A4B

```


5624 CDEB40	CALL #40E8	56A8 C9	RET
5627 B5	PUSH DE	56AB E5	PRVALU: PUSH HL
5628 CD482A	CALL #2A48	56AC 111E57	LD DE,NUMBUF
5629 7A	LD A,D	56AF 010500	LD BC,5
562C B7	DR A	56F2 D5	PUSH DE
562D C23914	JP NZ,#1439	56F3 E0B0	LDIR
5630 E3	IN (BP),HL	56F5 E1	POP HL
5631 DBE8	IN A,(#EB)	56F6 110D4A	LD DE,#4A0D
5633 C8F5	SET 7:A	56F9 D5	PUSH DE
5635 D3E8	OUT (#EB),A	56FA CD1831	CALL #3118
5637 73	LD (HL),E	56FD D1	POP DE
5638 DBE8	IN A,(#EB)	56FE C0B908	CALL #08B9
563A C8F5	REB 7:A	5701 E1	POP HL
563C D3E8	OUT (#EB),A	5702 C35718	JP #1857
563E C37422	JP #2274	5705 3E22	PRVALU: LD A,#22
5641	I	5707 CDB508	CALL #08B5
5641	IFOR VPEEK SUPPORT	570A 7E	LD A,(HL)
5641	I	570B 87	OR A
5641 CD6018	FR2560: CALL #1860	570C 2808	JR Z,PR*
5644 B1	DB #B1	570E 46	LD B,(HL)
5645 3314	DW #1433	570F 23	PRCON: INC HL
5647 23	INC HL	5710 7E	LD A,(HL)
5648 FE92	CP #92	5711 CDB508	CALL #08B5
564A DA3314	JP C,#1433	5714 10F9	DJNZ PRCON
564D E5	PUSH HL	5716 23	PR*: INC HL
564E 210A53	LD HL,EXETBL	5717 23	INC HL
5651 C3661C	JP #1C66	5718 3E22	LD A,#22
5654	I	571A CDB508	CALL #08B5
5654	IVLIST	571D C9	RET
5654	I	571E	NUMBUF: DS 5
5654 CD061A	VLIST: CALL #1A06	5723	I
5657 200C	JR NZ,7VLNUM	5723	DELETE
5659 CDB456	CALL VLISTN	5723	I
565C CD2E0A	CALL #0A2E	5723 CD061A	DELETE: CALL #1A06
565F CD9956	CALL VLISTS	5726 2003	JR NZ,7DELN
5662 C5FF1B	CP #1BFF	5728 C33314	JP #1433
5665 FE4E	7VLNUM: CP "N"	572B E5	7DELN: PUSH HL
5667 200C	JR NZ,7VLBTR	572C 21B34B	LD HL,#4BB3
5669 23	INC HL	572F CDB41A	CALL #1AB4
566A CD061A	CALL #1A06	5732 3D	DEC A
566D C23314	BYNER: JP NZ,#1433	5733 CDB51A	CALL #1AB5
5670 CDB456	CALL VLISTN	5736 E1	POP HL
5673 18ED	JR 7VLNUM-3	5737 CDC21D	CALL #1DC2
5675 FE53	CP "B"	573A 2AB54B	LD HL,#4AB5
5677 20F4	JR NZ,BYNER	573B EDBB34B	LD DE,(#4BB3)
5679 23	INC HL	5741 CDB318	CALL #1B3E
567A CD061A	CALL #1A06	5744 2866	JR Z,DELN
567D 20E6	JR NZ,BYNER	5746 3801	JR C,7DELNM
567F CD9956	CALL VLISTS	5748 EB	EX DE,HL
5682 18DE	JR 7VLNUM-3	5749 7C	7DELNM: LD A,H
5684 E5	VLISTN: PUSH HL	574A B5	OR L
5685 2ABA4F	LD HL,(#4FBA)	574B 2005	JR NZ,7DELNM
5688 CDB856	CALL NOCHK:	574D 2106A0	LD H,6000
568B E1	POP HL	5750 1B3F	7DELNM+7
568C CB	RET Z	5752 7A	LD A,D
568D E5	PUSH HL	5753 A3	AND E
568E CDB756	CALL VALNP	5754 3C	INC A
5691 CDB856	CALL PRVALU	5755 2033	JR NZ,7DELNM
5694 CDAE56	CALL BRKSP	5757 CD7B19	DELN-1: CALL #197B
5697 18EF	JR VLISTN+4	575A 5D14	DW #145D
5699 E5	VLISTS: PUSH HL	575C EB	EX DE,HL
569A 2A904F	LD HL,(#4F90)	575D 2A4E4F	LD HL,(#4F4E)
569D CDB856	CALL NOCHK:	5760 2B	DEC HL
56A0 E1	POP HL	5761 2B	DEC HL
56A1 CB	RET Z	5762 D5	PUSH DE
56A2 E5	PUSH HL	5763 EB	EX DE,HL
56A3 CDD156	CALL VALSP	5764 CD5418	CALL #1854
56A6 CD0557	CALL PRSVAL	576F ED43C557	CALL BC+HSD
56A9 CDAE56	CALL BRKSP	5768 EB	LD (DE,NUM),BC
56AC 18EF	JR VLISTB+4	576B D1	EX DE,HL
56AE CDA205	CALL #05A2	576F D1	POP DE
56B1 CAF612	JP Z,#12F6	5770 CDBE57	CALL BC+HSD
56B4 DBEA	STOP: IN A,(#EA)	5773 C5	PUSH BC
56B6 FEFD	CP #FD	5774 ED4BC557	LD BC,(DE,NUM)
56B8 28FA	JR Z,STOP	5779 B1	LD A,B
56BA C9	RET	577A 2808	OR C
56BB 7E	NOCHK: LD A,(HL)	577C D5	JR Z,7DELNM-3
56BC 23	INC HL	577D E0B0	PUSH DE
56BD B6	DR (HL)	577F D1	LDIR
56BE CB	RET Z	5780 C1	POP DE
56BF 2B	DEC HL	5781 CD2D18	POP BC
56C0 F5	PUSH AF	5784 CDB213	CALL #182D
56C1 CD2E0A	CALL #0A2E	5787 C3F612	CALL #13F2
56C4 F1	POP AF	578A D5	JP #12F6
56C5 EB	EX DE,HL	578B CD7B19	DELN-1: PUSH DE
56C6 C9	RET	578E 5D14	CALL #197B
56C7 EB	EX DE,HL	5790 D1	DW #145D
56C8 CDBC56	CALL PRVNUM	5791 E5	POP DE
56CB 3E3D	LD A,""	5792 EB	PUSH HL
56CD CDB508	CALL #08B5	5793 CD7B19	EX DE,HL
56D0 C9	RET	5796 5D14	CALL #197B
56D1 EB	EX DE,HL	5798 2004	DW #145D
56D2 CDDC56	JR VALNP+4	579A 7E	JR NZ,7DELNM1
56D5 3E24	LD A,"8"	579B 23	LD A,(HL)
56D7 CDB508	CALL #08B5	579C 66	INC HL
56DA 18EF	JR VALNP+4	579D 6F	LD H,(HL)
56DC 7E	PRVNUM: LD A,(HL)	579E E5	LD A
56DD CDB508	CALL #08B5	579F CD5418	DELNM1: PUSH HL
56E0 23	INC HL	57A2 D1	CALL #1B54
56E1 7E	LD A,(HL)	57A3 C9	POP DE
56E2 FE20	CP #20	57A4 CDBE57	CALL BC+HSD
56E4 280A3	JR Z,PRVRET	57A6 ED43C557	LD (DE,NUM),BC
56E6 CDB508	CALL #08B5	57AA 18C2	JR DELN-1
56E9 23	PRVRET: INC HL	57AC CDE614	DELN: CALL #14E6

BASIC拡張プログラム・リスト

```

57AF CDE114 CALL #14E1
57B2 CD7819 CALL #1478
57B5 5D14 DN #145D
57B7 20CE JR NZ,DELN-M-3
57B9 CCAB13 JR DELN-M-3
57BC 18C9 LD A,L
57BE 7D BCH:HSO1 LD A,L
57BF 93 SUB E
57C0 4F LD C,A
57C1 7C LD A,H
57C2 9A SBC A,D
57C3 47 LD B,A
57C4 C9 RET
57C5 0000 DELNUM DN 0
57C7 I
57C7 I WHILE INITIAL
57C7 I
57C7 FR12FA1 LD HL,WE2AK
57CA 22BC58 LD (WSPINT),HL
57CD AF XOR A
57CE 32BA58 LD (WCOUNT),A
57D1 110C14 LD DE,#140C
57D4 C3F012 JP #12FD
57D7 I
57D7 I WHILE
57D7 I
57D7 WHILE: PUSH HL
57D8 2B DEC HL
57D9 2B DEC HL
57DA 225151 LD (#5151),HL
57DD E1 POP HL
57DE C0A18B CALL #180A
57E1 E5 PUSH HL
57E2 21BA58 LD HL,WCOUNT
57E5 7E LD A,(HL)
57E6 FE10 CP 16
57E8 024E14 JC NC,#144E
57EB 34 JP INC (HL)
57EC E1 POP HL
57ED C06523 CALL #2365
57F0 C0F119 CALL #19F1
57F3 0B2A944F LD 11,(#4F94)
57F7 00C0A47E BIT 7,(11+4)
57FB 2B15 JR Z,EXET
57FD E5 PUSH HL
57FE 215251 LD HL,#5152
5801 E058BC58 LD DE,(WSPINT)
5805 010600 BC 6
5808 E0B8 L11 LDDR
580A E8 EX DE,HL
580B 22BC58 LD (WSPINT),HL
580E E1 POP HL
580F C3FF1B EXET: PUSH HL
5812 E5 PUSH HL
5813 21BA58 LD HL,WCOUNT
5816 35 DEC (HL)
5817 AF SERMEN: XOR A
5818 32B058 LD (WNUM),A
581B E1 POP HL
581C 7E LD A,(HL)
581D FE0D CP #0D
581F 2011 JR NZ,NEXCHR
5821 CD7558 CALL ENDCKW
5824 CA61D JR Z,#1DE6
5827 23 NXLTLN: INC HL
5828 114D51 LD DE,#514D
582B 010400 LD BC,4
582E EDB0 LDIR
5830 18EA JR SERMEN+5
5832 FE22 NEXCHR: #22
5834 23 INC HL
5835 282B JR Z,INCAD
5837 FEB1 CP #B1
5839 20E1 JR NZ,SERMEN+5
583B 7E LD A,(HL)
583C 23 INC HL
583D FEB3 CP #B3
583F 2008 JR NZ,WEND
5841 E5 PUSH HL
5842 21B858 LD HL,WNUM
5845 34 INC (HL)
5846 E1 POP HL
5847 18D3 JR SERMEN+5
5849 FEB4 WEND: CP #B4
584B 20CF JR NZ,SERMEN+5
584D 34B858 LD A,(WNUM)
5850 B7 DR A
5851 2806 JR Z,FWEND
5853 3D DEC A
5854 32B858 LD (WNUM),A
5857 18C3 JR SERMEN+5
5859 225151 FWEND: LD (#5151),HL
585C 210000 LD HL,0
585F C3FC1B JP #1BFC
5862 7E INCAD: LD A,(HL)
5863 23 INC HL
5864 FE22 CP #22
5866 28B4 JR Z,SERMEN+5
5868 7E LD A,(HL)
5869 FE0D CP #0D
586B 28BA JR Z,NXLTLN
586D CD7558 CALL ENDCKW
5870 CA61D JP Z,#1DE6
5873 18ED JR INCAD

```

```

5875 E3 ENDCKW: PUSH HL
5876 2A0451 LD HL,(#514D)
5879 7C LD A,H
587A B5 OR L
587B CAF612 JP Z,#12F6
587C 7E LD A,(HL)
587F 23 INC HL
5880 BA OR (HL)
5881 E1 POP HL
5882 C9 RET
5883 I
5883 I MEND
5883 I
5883 TABA58 WEND: LD A,(WCOUNT)
5886 B7 OR A
5887 CA5414 JP Z,#1454
588A 3D DEC A
588B 32BA58 LD (WCOUNT),A
588C 2ABC58 LD HL,(WBPINT),A
5891 114D51 LD DE,#514D
5894 010600 LD BC,6
5897 23 INC HL
5898 EDB0 LDIR
589A 2B DEC HL
589B 22BC58 LD (WSPINT),HL
589E 2A5151 LD HL,(#5151)
58A1 C3231C JP #1C23
58A4 I
58A4 I 17*17 / 7
58A4 I
58A4 FR14F5: CALL #18C1
58A7 FE20 CP #20
58A9 C0 RET NZ
58AA 23 INC HL
58AB C9 RET
58AC 110000 FR18A0: LD DE,0
58AF C0A18B CALL #180A
58B2 C9 RET
58B3 7E FR18A4: LD A,(HL)
58B4 D630 SUB #30
58B6 FE0A CP #0A
58B8 7E LD A,(HL)
58B9 C9 RET
58BA 00 WCOUNT: DB 0
58BB 00 WNUM: DB 0
58BC 0000 WSPINT: DW 0
58BE 00 DB 99
58C1 00 WESTAK: DB 0
58C2 1XXXXXXX XXXXXXXXXXXXXXXX
58C2 I BASIC USING PROCESB
58C2 I ORG=#5922
58C2 I PRINT USING"###.##"
58C2 1XXXXXXXXXXXXXXXXXXXXX
58C2 BLANK: EQU #1809
58C2 ERDIP: EQU #3A6B
58C2 ERDISP: EQU #1433
58C2 USING: CALL #1A06 FROM #3A5F
58C5 CAFE3A JP Z,#3A5F
58C8 E5 PUSH HL
58C9 FE3B CP #3B
58CB 2804 JR Z,7D
58CD FE2C CP #2C
58CF 2001 JR NZ,7D+1
58D1 23 INC HL
58D2 C0A18B CALL #180A
58D3 B1 DB #B1
58D5 3C59 DW N09
58D7 FEB6 CP #B6
58DA 2804 JR Z,3PROC
58DB E1 POP HL
58DD C3AB3A N09: JP END9
58DE F1 3PROC: POP AF
58DF CD0918 CALL BLANK
58E1 FE22 CP #22
58E4 2074 JR NZ,INIT?
58E6 CD0918 CALL BLANK
58E8 0E00 LD C,0
58EA 7E LD A,(HL)
58EC FE23 CP #E
58ED 2004 JR NZ,LBL1
58EE 0C INC C
58EF 23 INC HL
58F0 18F7 JR LBL6
58F2 0B LBL1: EX AF,AF'
58F3 79 LD A,C
58F5 B7 OR A
58F7 CADC59 JP Z,USNGER
58F9 3D A DEC A
58FB 32E159 LD (INTPT),A
58FC 0B EX AF,AF'
58FE FE2E CP #E
5901 2033 JR NZ,LBL2
5903 32E459 LD (PINTF),A
5905 0E00 LD C,0
5907 23 INC HL
5909 7E LBL5: LD A,(HL)
590B FE23 CP #E
590D 2004 JR NZ,LBL3
590F 0C INC C
5911 23 INC HL
5913 18F7 JR LBL5
5915 0B LBL3: EX AF,AF'
5917 79 LD A,C
5919 32E259 LD (FRACPT),A

```

BASIC拡張プログラム・リスト

5979 08	EX AF, AF	5A51 12	LD (DE), A
597A FE22	CP #22	5A52 13	INC DE
597C 202D	JP NZ, LBL4	5A53 0A1B	LD B, 27
597E AF	XOR A	5A53 3E30	LD A, #30
597F 32E359	LD (EXPR1), A	5A57 12	LBL52: LD (DE), A
5982 23	INC HL	5A58 13	INC DE
5983 E5	PUSH HL	5A59 10FC	DJNZ LBL52
5984 21E159	LD HL, INTPT	5A5B 3E0D	LBL53: LD A, #0D
5987 7E	LD A, (HL)	5A5D 12	LD (DE), A
598B 23	INC HL	5A5C 3AE359	LBL04: LD A, (EXPR1)
598F 06	ADD A, (HL)	5A61 B7	OR A
599A E1	POP HL	5A62 C23B5B	JP NZ, EXPONE
599B FE1D	CP 27	5A65 3AE159	LD A, (INTPT)
599D 02DC59	JP NC, USNGER	5A6B 47	LD B, A
599E 3E01	LD A, 1	5A69 E1	POP HL
599F 32E559	LD (3FLAG), A	5A6B 0E00	PUSH HL
5995 C36B3A	JP END3	5A4D 7E	LBL22: LD A, (HL)
599B 08	LBL2: EX AF, AF	5A4E FE2E	CP "
5999 AF	XOR A	5A50 2B05	JP Z, L23
599A 32E259	LD (FRACPT), A	5A52 23	INC HL
599D 32E459	LD (POINT1), A	5A53 0C	INC C
59A0 08	EX AF, AF	5A54 C34D5A	JP LBL22
59A1 FE22	CP #22	5A57 79	LD A, C
59A1 2006	JP NZ, LBL4	5A5B FE02	CP 2
59A5 AF	XOR A	5A5A 2015	JP NZ, LBL23
59A6 32E359	LD (EXPR1), A	5A5C 2B	DEC HL
59A9 18D7	LD LBL7	5A5D 7E	LD A, (HL)
59AB FE5E	LBL4: CP	5A5E FE30	CP #30
59AD 201B	LD LBL0	5A60 200F	JP NZ, LBL23
59AF 32E359	LD (EXPR1), A	5A62 7B	LD A, (HL)
59B2 CD091B	CALL BLANK	5A63 B7	OR A
59B5 FE22	CP #22	5A64 200B	JP NZ, LBL23
59B7 20F4	JP NZ, LBLB	5A66 210D4A	LD HL, #4AD0
59B9 C3B259	JP LBL7	5A69 7E	LD A, (HL)
59BC FE1B	QINIT7: CP "	5A6A 23	INC HL
59BE 2B0D	JP Z, INIT	5A6B 77	LD (HL), A
59C0 FE2C	CP "	5A6C 117F5C	LD DE, PRBUF
59C2 2B09	JP Z, INIT	5A6F 1B15	JP LBL25
59C4 FE0D	CP #0D	5A71 217F5C	LBL23: LD A, PRUF
59C6 2B05	JP Z, INIT	5A74 0D	DEC C
59C8 FE3A	CP "	5A75 7B	LD A, B
59CA C2DC59	LBL0: JP NZ, USNGER	5A76 91	SUB C
59CD AF	INIT: XOR A	5A77 DA7A5C	JP Z, 30TAEH
59CE E5	PUSH HL	5A7A 2B07	LD B, A
59CF 21E159	LD HL, INTPT	5A7C 47	LD A, #20
59D2 0605	LD B, 5	5A7D 3E20	LD (HL), A
59D4 77	LD (HL), A	5A7F 77	INC HL
59D5 23	INC HL	5A80 23	DJNZ LBL24
59D6 10FC	DJNZ LBL9	5A81 10FC	POP DE
59D8 E1	POP HL	5A83 D1	LBL07: PUSH DE
59D9 C36B3A	JP END3	5A84 D5	EX DE, HL
59DC 3E14	USNGER: LD A, 20	5A85 EB	LDI
59DE C3B814	JP #148B	5A86 EDA0	LBL25: LD A, (HL)
59E1 00	INTPT: DB 0	5A8B 7E	CP "
59E2 00	FRACPT: DB 0	5A89 FE2E	JP NZ, LBL25
59E3 00	EXPR1: DB 0	5A8B C2B65A	CP "
59E4 00	POINT1: DB 0	5A8E EDA0	LDI
59E5 00	3FLAG: DB 0	5A90 3AE259	LD A, (FRACPT)
59E6	XXXXXXXXXXXXXXXXXXXXXXXXXXXX	5A93 B7	OR A
59E6	1 NUMERIC OUTPUT IN USING	5A94 200A	JP NZ, FRACLD
59E6	1 ORG #59E6	5A9A 3AE459	LD A, (POINT1)
59E6	XXXXXXXXXXXXXXXXXXXXXXXXXXXX	5A99 B7	OR A
59E6	FRACFF: CALL #311B	5A9A 2002	JP NZ, FRACLD-2
59E9 E1	POP HL	5A9C 2B	DEC HL
59EA AF	XOR A	5A9D 1B	DEC DE
59EB 329F5C	LD (PNTFLO), A	5A9E 1B05	FRACLD: LD B, C
59EE 329D5C	LD (EXP1), A	5A9A 0600	LD C, A
59F1 3AE559	LD A, (3FLAG)	5A92 4F	LDIR
59FA B7	OR A	5A93 ED8B	WCR: LD A, #0D
59FB C2F859	JP NZ, 3PRINT	5A95 3E0D	LD (DE), A
59FB C3033B	JP #3B03	5A97 12	POP DE
59FB E5	3PRINT: PUSH HL	5A9B D1	PUSH DE
59FC 7E	LD A, (HL)	5A99 D5	LD HL, PRBUF
59FD 23	INC HL	5A9A 217F5C	LD B, 29
59FE FE0D	CP #0D	5A9D 10D00	LDIR
5A00 CAB65A	JP Z, LBL02	5A9B ED8B	POP HL
5A03 FE45	CP "E"	5A9C 3C033B	JP #3B03
5A05 C2FC59	JP NZ, 3PRINT+1	5A9B	:
5A0B 7E	LD A, (HL)	5A96 E1	LBL02: POP HL
5A09 FE2B	CP "	5A97 E5	LD A, (HL)
5A0B C2F35A	JP NZ, LBL03	5A9B 7E	LBL26: LD

BASIC拡張プログラム・リスト

5A0B 3803	JR C,LBL60	5B77 F2875B	JP P,LBL74
5A0D C3E25A	JP LBL29	5B7A 2F	CPL
5A0E 3E30	LD A,*30	5B7B C601	ADD A,1
5A0E 77	LD (HL),A	5B7D 4F	LD C,A
5A0E 23	INC HL	5B7E 54	LD D,H
5A0E 10EA	DJNZ LBL30	5B7F 5D	LD E,L
5A0E 180A	JR WSPC+6	5B80 EB	DEC HL
5A0E 3E30	LD A,*30	5B81 2B	EXC HL
5A0E 77	LD (HL),A	5B82 EDB8	LDDR
5A0E 23	INC HL	5B8A 23	INC HL
5A0E 10FC	DJNZ WSPC+2	5B8B 1907	JR LBL75
5A0E 00	NOP	5B87 4F	LD C,A
5A0E 00	NOP	5B8B 54	LD D,H
5A0F C33E5A	JP LBL04	5B89 5D	LD E,L
5AF3		5B8A 23	INC HL
5AF3 CD425B	CALL ASIBIN	5B8B EDB0	LDIR
5AF6 E1	POP HL	5B8D 2B	DEC HL
5AF7 E5	PUSH HL	5B8E 3E2E	LD A,*
5AF8 7E	LD A,(HL)	5B90 77	LD (HL),A
5AF9 FE45	CP "E"	5B91 C3175C	JP LBL81
5AFB 2B04	JR Z,LBL32	5B94 2B	DEC HL
5AFD 23	INC HL	5B95 7E	LD A,(HL)
5AFE C3F85A	JP LBL31	5B96 FE30	CP *30
5B01 2B	DEC HL	5B9B 23	INC HL
5B02 54	LD D,H	5B99 20D0	JR NZ,LBL77
5B03 5D	LD E,L	5B9B E5	PUSH HL
5B04 3A9D5C	LD A,(EXP)	5B9C 23	INC HL
5B07 3C	INC A	5B9D 0E00	LD C,0
5B0B 4F	LD C,A	5B9F 7E	LD A,(HL)
5B09 7B	LD A,E	5BA0 FE30	CP *00
5B0A 81	ADD A,C	5BA2 2004	JR NZ,LBL7A
5B0B 5F	LD E,A	5BA4 23	INC HL
5B0C 7A	LD A,D	5BA5 0C	INC C
5B0D CE00	ADC A,0	5BA6 1BF7	JR LBL7B
5B0F 57	LD D,A	5BA8 D1	POP DE
5B10 C1	POP BC	5BAF 329C5C	LD (ZEROF),A
5B11 C5	PUSH BC	5BAC FE0D	CP *00
5B12 E5	PUSH HL	5BAE 3AE159	LD A,(INTPR)
5B13 B7	OR A	5BB1 F5	PUSH AF
5B14 ED42	SEC HL,BC	5BB2 2006	JR NZ,LBL150
5B16 44	LD B,H	5BB4 AF	XOR A
5B17 4D	LD C,L	5BB5 329E5C	LD (EXPVAL),A
5B1B E1	POP HL	5BBB 1804	JR LBL151
5B19 0B	DEC BC	5BB8 01	LD A,C
5B1A 7B	LD A,B	5BB9 329E5C	LD (EXPVAL),A
5B1B B1	OR C	5BBE F1	POP AF
5B1C CA7A5C	JP Z,DTAER	5BBF 87	OR A
5B1F 13	INC DE	5BC0 201A	JR NZ,LBL79
5B20 3E0D	LD A,*0D	5BC2 3E2E	LD A,*
5B22 12	LD (DE),A	5BC4 1B	DEC DE
5B23 1B	DEC DE	5BC5 12	LD (DE),A
5B24 D5	PUSH DE	5BC6 13	INC DE
5B25 EDB8	LDDR	5BC7 3A9C5C	LD A,(ZEROF)
5B27 D1	POP DE	5BCA FE0D	CP *0D
5B2B E1	POP HL	5BCC 2005	JR NZ,LBL80
5B29 E5	PUSH HL	5BCE 3E30	LD A,*30
5B2A D5	PUSH DE	5BD0 12	LD (DE),A
5B2B 23	INC HL	5BD1 1844	JR LBL81
5B2C 3E30	LD A,*30	5BD3 EDA0	LDI
5B2E 77	LD (HL),A	5BD5 7E	LD A,(HL)
5B2F 25	INC HL	5BD6 FE0B	CP *0D
5B30 3E2E	LD A,*	5BD8 20F9	JR NZ,LBL80
5B32 77	LD (HL),A	5BDA 183B	JR LBL81
5B33 23	INC HL	5BDC 0B	DEC HL
5B34 3A9D5C	LD A,(EXP)	5BDD 3A9C5C	LD A,(ZEROF)
5B37 47	LD B,A	5BEO FE0D	CP *0D
5B3B 3E30	LD A,*30	5BE2 2B53	JR Z,LBL81
5B3A 77	LD (HL),A	5BE4 0B	EX AF,AF
5B3B 23	INC HL	5BE5 D5	PUSH DE
5B3C 10FC	DJNZ LBL34	5BE6 54	LD D,H
5B3E E1	POP HL	5BE7 5D	LD E,L
5B3F C3CD5A	JP LBL28	5BE8 1B	DEC DE
5B42 23	INC HL	5BE9 0600	LD C,A
5B43 C5	PUSH BC	5BED 4F	PUSH DE
5B44 0E00	LD C,0	5BED EDB0	LDDR
5B46 7E	LD A,(HL)	5BEF 2B	DEC HL
5B47 D630	SUB *30	5BF0 3E2E	LD A,*
5B49 2B03	JR Z,LBL40	5BF2 77	LD (HL),A
5B4B 3E0A	LD A,10	5BF3 E1	POP HL
5B4D 4F	LD C,A	5BF4 D1	POP DE
5B4E 23	INC HL	5BF5 1B	DEC DE
5B4F 7E	LD A,(HL)	5BF6 061B	LD B,27
5B50 D630	SUB *30	5BF8 7E	LD A,(HL)
5B52 B1	ADD A,C	5BF9 FE0D	CP *0D
5B53 329D5C	LD (EXP),A	5BFB 2013	JR NZ,LBL82B
5B56 C1	POP BC	5BFD 3E30	LD A,*30
5B57 C9	RET	5BFF 12	LD (DE),A
5B58 E1	POP HL	5C00 13	INC DE
5B59 E5	PUSH HL	5C01 23	INC HL
5B5A 0E00	LD C,0	5C02 7E	LD A,(HL)
5B5C 7E	LD A,(HL)	5C03 FE2E	CP *
5B5D FE2E	CP *	5C05 2B02	JR Z,LBL82D
5B5F 2B04	JR Z,LBL71	5C07 3E30	LD A,*30
5B61 23	INC HL	5C09 10F4	LD A,*30
5B62 0C	INC C	5C0B 3E0D	LD A,*30
5B63 1BF7	JR LBL70	5C0D 12	LD (DE),A
5B65 0B	LD A,C	5C0E 1807	JR LBL81
5B66 79	CP 2	5C10 12	LD (DE),A
5B67 FE02	JR C,LBL72	5C11 23	INC HL
5B69 3AE159	LD A,(INTPR)	5C12 13	INC DE
5B6E 91	SUB C	5C13 10E3	DJNZ LBL82A
5B6F 329E5C	LD (EXPVAL),A	5C15 1BF4	JR LBL82B-5
5B72 0A00	LD A,0	5C17 E1	POP HL
5B74 CA175C	JP Z,LBL81	5C18 E5	PUSH HL

BASIC拡張プログラム・リスト

SC19 7E	LD	A ₁ (HL)	SC54 F1	LBL87:	POP	AF
SC1A FE2E	CP	"."	SC55 23		INC	HL
SC1C 23	INC	HL	SC56 1600		LD	D,0
SC1D 20FA	JR	NZ,LBL81+2	SC58 D60A	LBL89:	SUB	10
SC1F 3AE259	LD	A ₁ (FRACPT)	SC5A 3B03		JR	C,LBL8B
SC22 B7	OR	A	SC5C 14		INC	B
SC23 200C	JR	NZ,LBL95	SC5D 1BF9		JR	LBL89
SC25 3AE459	LD	A ₁ (POINTF)	SC5F C60A	LBL88:	ADD	A,10
SC28 B7	OR	A	SC61 C630		ADD	A,#30
SC29 2001	JR	NZ,LBL83	SC63 8F		LD	E,A
SC2B 2B	DEC	HL	SC64 7A		LD	A,D
SC2C 3E45	LBL83:	LD A,""	SC65 C630		ADD	A,#30
SC2E 77	LD	(HL),A	SC67 77		LD	(HL),A
SC2F 1B07	JR	LBL84	SC68 23		INC	HL
SC31 0A00	LD	B,0	SC69 73		LD	(HL),E
SC33 4F	LBL85:	LD C,A	SC6A 23		INC	HL
SC34 09	ADD	HL,BC	SC6B 3E0D		LD	A,#0D
SC35 3E45	LD	A,""	SC6D 77		LD	(HL),A
SC37 77	LD	(HL),A	SC6E E1		POP	HL
SC3B 23	LBL84:	INC HL	SC6F C3033B	LBL90:	JP	#3B03
SC39 3AE5C	LD	A ₁ (EXPVAL)	SC72			
SC3C F5	PUSH	AF	SC72 CD421A		2FREST:	CALL #1A42
SC3D B7	OR	A	SC75 AF		XOR	A
SC3E FAAC5C	JP	HL,LBL86	SC7A 32E559		LD	(2FLA0),A
SC41 2004	JR	NZ,LBL170	SC79 C9		RET	
SC43 3E2B	LD	A,""	SC7A			
SC45 1B02	JR	LBL170+2	SC7A 3E15		DATAER:	LD A,21
SC47 3E2D	LD	A,""	SC7C C3B814			
SC49 77	LD	(HL),A	SC7F		PRBUF:	DB 29
SC4A 1B0B	JR	LBL87	SC7C		ZEROF:	DB 0
SC4C 3E2B	LBL86:	LD A,""	SC7D		EXP:	DB 0
SC4E 77	LD	(HL),A	SC7E		EXPVAL:	DB 0
SC4F F1	POP	AF	SC7F		INTFLG:	DB 0
SC50 2F	CPL		SC80			
SC51 C601	ADD	A,1	SCA0			
SC53 FE	PUSH	AF	SCA0			END

BASIC書き換えプログラム・リスト

7B30	*****		7BCE 22F712	LD	(#12F7),HL
7B30	I	インタプリタ カナヒ & SAVE	7BD1 213355	LD	HL,#5533
7B30	*****		7BD4 22571C	LD	(#1C57),HL
7B30	I		7BD7 214055	LD	HL,#5540
7B30		ORG #7B30	7BD8 22751F	LD	(#1F75),HL
7B30			7BD0 215355	LD	HL,#5553
7B30	I	カナヒプログラム #5160 の終了	7BE0 22CC22	LD	(#22CC),HL
7B30			7BE3 216595	LD	HL,#55A5
7B30 210070	LD	HL,#7000	7BE6 22901F	LD	(#1F9D),HL
7B33 114051	LD	DE,#5160	7BE9 217D55	LD	HL,#557D
7B36 01200B	LD	BC,#0B20	7BEC 224521	LD	(#2145),HL
7B39 ED80	LDIR		7BEF 218555	LD	HL,#55B5
7B3B	I		7BF2 22701F	LD	(#1F70),HL
7B3B	I	インタプリタ カナヒ	7BF5 219A55	LD	HL,#559A
7B3B			7BF8 227C19	LD	(#197C),HL
7B3B 21FF5F	LD	HL,#55FF	7BF8 214156	LD	HL,#5641
7B3C 22A912	LD	(#12A9),HL	7BFE 226125	LD	(#2561),HL
7B41 23	INC	HL	7C01 212599	LD	HL,#5592
7B42 227F19	LD	(#197F),HL	7C04 225B3A	LD	(#3A5B),HL
7B45 22221A	LD	(#1A22),HL	7C07 21E659	LD	HL,#55E6
7B48 22921B	LD	(#1B92),HL	7C0A 22003B	LD	(#3B00),HL
7B4B 22AC1B	LD	(#1BAC),HL	7C0D 21725C	LD	HL,#5C72
7B4E 22DB1B	LD	(#1BDB),HL	7C10 228B1F	LD	(#1F8B),HL
7B51 22E31B	LD	(#1BE3),HL	7C13 21AC58	LD	HL,#5BAC
7B54 225A1D	LD	(#1A5A),HL	7C16 22A11B	LD	(#1BA1),HL
7B57 22941F	LD	(#1F94),HL	7C19 21B358	LD	HL,#5B83
7B5A 22753C	LD	(#3C75),HL	7C1C 22A51B	LD	(#1BA5),HL
7B5D 22633D	LD	(#3D63),HL	7C1F 21A458	LD	HL,#5BA4
7B60 22653E	LD	(#3E65),HL	7C22 22F614	LD	(#1F61),HL
7B63 22A83E	LD	(#3E6B),HL	7C25 21C757	LD	HL,#5C7C
7B66 23	INC	HL	7C28 22F812	LD	(#12F8),HL
7B67 23	INC	HL	7C2B		
7B68 22B63D	LD	(#3D86),HL	7C2B		
7B6B 3EC9	LD	A,#C9	7C2B 21D757		LD HL,#5D7D
7B6D 32DF44	LD	(#44DF),A	7C2C 221053		LD (#5310),HL
7B70 216051	LD	HL,#5160	7C31 21B358		LD HL,#5B83
7B73 226113	LD	(#1361),HL	7C34 221253		LD (#5312),HL
7B76 219952	LD	HL,#5299	7C37 3E32		LD #32
7B79 223115	LD	(#1531),HL	7C39 32B813		LD (#13B8),A
7B7C 22BD15	LD	(#15BD),HL	7C3C 114111		LD DE,#1141
7B7F 210053	LD	HL,#5300	7C3F 21497C		LD HL,#5B52
7B82 22631C	LD	(#1C63),HL	7C42 010E00		LD BC,#0E
7B85			7C45 EDB0		LDIR
7B85 3E3C	LD	A,#C3	7C47 1B0E		JR JUMP
7B87 32AE00	LD	(#00AE),A	7C49 42415349	SB5522:	DN "BASIC 5B-5522"
7B8A 320A1A	LD	(#1A0A),A	7C4D 43205342		
7B8D 32FF1B	LD	(#1BFF),A	7C51 2B353532		
7B90 32AF22	LD	(#22AF),A	7C55 32		
7B93 32F612	LD	(#12F6),A	7C56 0D		DB #0D
7B96 32561C	LD	(#1C56),A	7C57		
7B99 32741F	LD	(#1F74),A	7C57		
7B9C 32CB22	LD	(#22CB),A	7C57		
7B9F 329C1F	LD	(#1F9C),A	7C57 210000	JUMP:	LD HL,#0
7BA2 324421	LD	(#2144),A	7C5A 1100B0		LD DE,#B000
7BA5 326F1F	LD	(#1F6F),A	7C5D 017F3C		LD BC,#5C7F
7BA8 327B19	LD	(#197B),A	7C60 EDB0		LDIR
7BAB 325A3A	LD	(#3A5A),A	7C62		
7BAE 32FF3A	LD	(#3AFF),A	7C62		
7BB1 32FA12	LD	(#12FA),A	7C62		
7BB4 3EC0	LD	A,#C0	7C62 217F5C		LD HL,#5C7C
7BB6 32A01B	LD	(#1B0A),A	7C65 225211		LD (#1152),HL
7BB9			7C6B 210000		LD HL,#0
7BB9			7C6B 225611		LD (#1156),HL
7BB9 214954	LD	HL,#5449	7C6E 2100B0		LD HL,#B000
7BBE 22071A	LD	(#1A07),A	7C71 225411		LD (#1154),HL
7BBF 213654	LD	HL,#5436	7C74 C05102		CALL #0251
7BC2 22001C	LD	(#1C00),HL	7C77 C0B202		CALL #02B2
7BC5 213053	LD	HL,#5330	7C7A C3AE00		JP #00AE
7BC8 22B022	LD	(#22B0),HL	7C7D		
7BCB 215B54	LD	HL,#545B	7C7D		END

日本語

エラー

U-Z

この「日本語エラー
メッセージ」が目に入らぬやうに



●佐々木雄一

MZ-80BのBASICはエラー表示がコードNo.で出てきますが、これが意外と不便で、たまにしか出ないものはそのつどマニュアルを引っ張り出して見なければなりません。80KのSP-5030あたりまではちゃんとメッセージで出ていたのですが、ディスクの6010からこの方式に変わっています。

というわけで、少しでもデバックの向上にとこのプログラムを作りました。

なお、SB-5520用なのであしからず、

1 使う前の注意

プログラムを打ち込む前にSB-5520を起動させて、MONでモニタへ移し、MコマンドがDコマンドで14B 2番地からの6バイトがそれぞれ、11、12、14、CD、B6、05であることを確認してください。もし違っているときはこのままでは動きません。

というのは、同じ5520でもバージョンが違うものがあるかもしれないからです。80Kのときは5030に新旧バージョンがありました。80Bの5520はそんな話は聞いていませんが、念のため書いておきます。

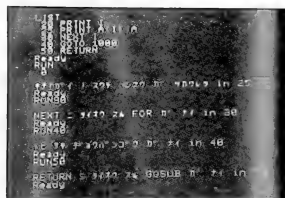
2 使い方

プログラムを打ち込んだら、テープにSAVEした後、F000からスタートしてください。SAVEするときにJ-ADRをF000にすれば、LOAD後オート・スタートします。その後モニタ・モードから自動的にBASICモードになりますから、LIMIT \$EFFFとしてください。これでもフリーエリアは40KB残ります。

あたりまえのことですが、F000からがこのプログラムのエリアなので、マシン語サブルーチンがこのへんにあるプログラムとはいっしょに使えません。ド手をするとき注意しますから充分注意してください。

これで使えるようになってるので、何かコマンドでない適当な文字をタイプして **CR** すると、ちゃんと“ブンポウ ジョウノ マチガイ”と出ますね？もし出なかったり、暴走したら、プログラムが違っているか、インタプリタのバージョンが異なるかのどちらかです。プログラム・モードのときは“in X”と行番号が後に付きます。

メッセージの内容はマニュアルの表現に従いましたが、なるべくわかりやすい表現にしました。エラーNo. だけよ



りも、さらに英語のメッセージよりもはるかに理解しやすいと思います。

3 プログラム

プログラムのほとんどがメッセージ・データです。エラーのときにHLレジスタのLに1バイトのエラーコードが入るので、これをアキュムレータに移し、比較して指定のアドレスへジャンプし、メッセージを表示させています。

アセンブル・リストを付けたので、メッセージの内容を変えたときは好きな順序にできます。アセンブラを持っている人は自分の好きなアドレスへリロートできますが、リストのF02B番地のLD(HL), F0HのF0はリロートする先頭アドレスの2バイトに変えてください(9000をスタート・アドレスにしたら90Hにする)。これを忘れるとインターリタから変なアドレスへジャンプしてしまいます。

4 おわりに

このプログラムはSB-5520用に作りましたが、はじめに説明したようにバージョンが違おうとうまく行きません。しかし、逆アセンブルしてエラー表示をしているところを探して書き換えれば、他のバージョンでも使えるようになります。

注) 12AC番地のFFをFEにしないと BASIC をコールド・スタートしたときにメッセージ・データはクリアされません。

** Z80 ASSEMBLER SB-7201 <N-ERRMSG> PAGE 01

08.17.82

```

01 F000                                ORG    F000H
02 F000                                :
03 F000 P          BASIC: EDU    1300H
04 F000 P          MSG:   EDU    0889H
05 F000 P          RET:   EDU    14BEH
06 F000                                : BASIC インタプリタ / カサカサ
07 F000 21B214                        LD      HL, 14B2H
08 F003 3600                          LD      (HL), 00H
09 F005 21B314                        LD      HL, 14B3H
10 F008 3600                          LD      (HL), 00H
11 F00A 21B414                        LD      HL, 14B4H
12 F00D 3600                          LD      (HL), 00H
13 F00F 21B514                        LD      HL, 14B5H
14 F012 3600                          LD      (HL), 00H
15 F014 21B614                        LD      HL, 14B6H
16 F017 3600                          LD      (HL), 00H
17 F019 21B714                        LD      HL, 14B7H
18 F01C 3600                          LD      (HL), 00H
19 F01E 21B814                        LD      HL, 14B8H
20 F021 3603                          LD      (HL), 03H
21 F023 21B914                        LD      HL, 14B9H
22 F026 363F                          LD      (HL), 3FH
23 F02B 21BD14                        LD      HL, 14BDH
24 F02B 36F0                          LD      (HL), F0H
25 F02D 21B214                        LD      HL, 14B2H
26 F030 36CD                          LD      (HL), CDH
27 F032 21B314                        LD      HL, 14B3H
28 F035 362E                          LD      (HL), 2EH
29 F037 21B414                        LD      HL, 14B4H
30 F03A 360A                          LD      (HL), 0AH
31 F03C C30013                        JP      BASIC
32 F03F                                ;return to basic
33 F03F 7D          ;15 コード 0000 jump
34 F040 FE01                        LD      A, L
35 F042 CAC0F0                        CP      Z, ER01
36 F045 FE02                        CP      02H
37 F047 CAC9F0                        JP      Z, ER02
38 F04A FE03                        CP      03H
39 F04C CAD2F0                        JP      Z, ER03
40 F04F FE04                        CP      04H
41 F051 CADBF0                        JP      Z, ER04
42 F054 FE05                        CP      05H
43 F056 CAE4F0                        JP      Z, ER05
44 F059 FE06                        CP      06H
45 F05B CAEDF0                        JP      Z, ER06
46 F05E FE07                        CP      07H
47 F060 CAF6F0                        JP      Z, ER07
48 F063 FE08                        CP      08H
49 F065 CAFFF0                        JP      Z, ER08
50 F068 FE0A                        CP      0AH
51 F06A CA0BF1                        JP      Z, ER0A
52 F06D FE0B                        CP      0BH
53 F06F CA11F1                        JP      Z, ER0B
54 F072 FE0C                        CP      0CH
55 F074 CA1AF1                        JP      Z, ER0C
56 F077 FE0D                        CP      0DH
57 F079 CA23F1                        JP      Z, ER0D
58 F07C FE0E                        CP      0EH
59 F07E CA2CF1                        JP      Z, ER0E
60 F081 FE0F                        CP      0FH

```

** Z80 ASSEMBLER SB-7201 <N-ERRMSG> PAGE 02

08.17.82

```

01 F083 CA35F1                        JP      Z, ER0F
02 F086 FE10                        CP      10H
03 F088 CA3EF1                        JP      Z, ER10
04 F08B FE11                        CP      11H
05 F08D CA47F1                        JP      Z, ER11
06 F090 FE12                        CP      12H
07 F092 CA50F1                        JP      Z, ER12
08 F095 FE13                        CP      13H
09 F097 CA59F1                        JP      Z, ER13
10 F09A FE18                        CP      18H

```

アセンブル・リスト

```

11 F09C CA62F1      JP      Z,ER18
12 F09F FE2B        CP      2BH
13 F0A1 CA6BF1      JP      Z,ER2B
14 F0A4 FE7F        CP      3FH
15 F0A6 CA74F1      JP      Z,ER3F
16 F0A9 FE41        CP      41H
17 F0AB CA7DF1      JP      Z,ER41
18 F0AE FE42        CP      42H
19 F0B0 CA86F1      JP      Z,ER42
20 F0B3 FE43        CP      43H
21 F0B5 CABFF1      JP      7,ER43
22 F0B8 FE46        CP      46H
23 F0BA CA9BF1      JP      Z,ER46
24 F0BD C30013      JP      BASIC
25 F0C0
26 F0C0 11A1F1      LD      DE,SYNTAX
27 F0C3 CDB908      CALL   MSG
28 F0C6 C3BE14      JP      RET
29 F0C9 11B5F1      LD      DE,C0VFI
30 F0CC CDB908      CALL   MSG
31 F0CF C3BE14      JP      RET
32 F0D2 11CEFF1     LD      DE,HENSHI
33 F0D5 CDB908      CALL   MSG
34 F0D8 C3BE14      JP      RET
35 F0DB 11E9F1     LD      DE,MISHT
36 F0DE CDB908      CALL   MSG
37 F0E1 C3BE14      JP      RET
38 F0E4 1104F2     LD      DE,STRING
39 F0E7 CDB908      CALL   MSG
40 F0EA C3BE14      JP      RET
41 F0ED 1121F2     LD      DE,MOVFI
42 F0F0 CDB908      CALL   MSG
43 F0F3 C3BE14      JP      RET
44 F0F6 117AF2     LD      DE,ARRY
45 F0F9 CDB908      CALL   MSG
46 F0FC C3BE14      JP      RET
47 F0FF 115AF2     LD      DE,LOVFL
48 F102 CDB908      CALL   MSG
49 F105 C3BE14      JP      RET
50 F108 1170F2     LD      DE,GSENST
51 F10B CDB908      CALL   MSG
52 F10E C3BE14      JP      RET
53 F111 1189F2     LD      DE,F0RNST
54 F114 CDB908      CALL   MSG
55 F117 C3BE14      JP      RET
56 F11A 11A3F2     LD      DE,DEFNST
57 F11D CDB908      CALL   MSG
58 F120 C3BE14      JP      RET
59 F123 118BF2     LD      DE,NOFOR
60 F126 CDB908      CALL   MSG

```

** Z80 ASSEMBLER SB-7201 -N-ERRMSG- PAGE 03

08.17.82

```

61 F129 C3BE14      JP      RET
62 F12C 11D4F2     LD      DE,N0G0SB
63 F12F CDB908      CALL   MSG
64 F132 C3BE14      JP      RET
65 F135 11F1F2     LD      DE,NODEF
66 F138 CDB908      CALL   MSG
67 F13B C3BE14      JP      RET
68 F13E 110EFF3    LD      DE,NOLINE
69 F141 CDB908      CALL   MSG
70 F144 C3BE14      JP      RET
71 F147 1125F3     LD      DE,N0CONT
72 F14A CDB908      CALL   MSG
73 F14D C3BE14      JP      RET
74 F150 11A8F3     LD      DE,WBASIC
75 F153 CDB908      CALL   MSG
76 F156 C3BE14      JP      RET
77 F159 11A4F3     LD      DE,DIRECT
78 F15C CDB908      CALL   MSG
79 F15F C3BE14      JP      RET
80 F162 11BDF3     LD      DE,N0DATA
81 F165 CDB908      CALL   MSG
82 F168 C3BE14      JP      RET

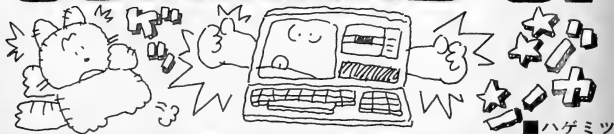
```


23 F16B 11A7F3	ER3B:	LD	DE,DOPEN
24 F16E 0D8908		CALL	MSG
25 F171 07BE14		JP	RET
26 F174 11CAE3	ER3F:	LD	DE,OUTFIL
27 F177 0D8908		CALL	MSG
28 F17A 07BE14		JP	RET
29 F17D 11DFF3	ER41:	LD	DE,PRIOFF
30 F180 0D8908		CALL	MSG
31 F183 07BE14		JP	RET
32 F186 1107F4	ER42:	LD	DE,PRTHA
33 F189 0D8908		CALL	MSG
34 F18C 07BE14		JP	RET
35 F18F 1110F4	ER43:	LD	DE,PRTHS
36 F192 0D8908		CALL	MSG
37 F195 07BE14		JP	RET
38 F198 1112F4	ER46:	LD	DE,SUMERR
39 F19B 0D8908		CALL	MSG
40 F19E 07BE14		JP	RET
41 F1A1	: ノキシー データ		
42 F1A1 07BE0BCF	Syntax:	DEFB	"7" 0000 0000 0000 0000
43 F1A5 0F720A0			
44 F1A9 0FAEB70			
45 F1AB 09200FC1			
46 F1B1 06FEB2			
47 F1B4 0D			
48 F1B5 040D8BDE	COVFL:	DEFB	0DH
49 F1B7 0D20B96F		DEFB	"123456789012345678901234567890"
50 F1BD 0620B6DE			
51 F1C1 20B92B0A			
52 F1C5 0E20C0DB			
53 F1C9 2070B0C0			
54 F1CC 0D			
55 F1CE 07C7B2B6	HEXUS:	DEFB	0DH
56 F1D2 0EB220C9		DEFB	"0123456789012345678901234567890"
57 F1D6 20B0B7C1			
58 F1DA 20C0D0B0			
59 F1DE 0720B6DE			
60 F1E2 20C2B6DC			
61 F1E6 DACC			
62 F1E8 0D			
63 F1E9 C7DE20C0	NISHT:	DEFB	0DH
64 F1ED 20C420CD		DEFB	"0123456789012345678901234567890"
65 F1F1 0D80B720			
66 F1F5 0920B6C0			
67 F1F9 20B6DE20			
68 F1FD B2AFC1BC			
69 F201 C5B2			
70 F203 0D			
71 F204 8DC40BDD	STRING:	DEFB	0DH
72 F208 8BDE20C7		DEFB	"0123456789012345678901234567890"
73 F20C 0E20C020			
74 F210 06DE2032			
75 F214 7575D7BC			
76 F218 DE20A620			
77 F21C 0AB400C7			
78 F220 0D			
79 F221 D2D7DB2D	MOVFL:	DEFB	0DH
80 F225 20C920D6		DEFB	"0123456789012345678901234567890"
81 F229 B7DBAEB7			
82 F22D 20B6DE20			
83 F231 00DB05B8			
84 F235 20C5AFC0			
85 F239 0D			
86 F23A CAB2DAC2	ARRY:	DEFB	0DH
87 F23E 20C920BE		DEFB	"0123456789012345678901234567890"
88 F242 0DB9DEDD			
89 F246 20B6DE20			
90 F24A 0FB420C9			
91 F24E 20D7C920			
92 F252 06DB20B5			
93 F256 B5B7B2			
94 F259 0D			
95 F25A 51B7DEAE	LOVFL:	DEFB	0DH
		DEFB	"0123456789012345678901234567890"



MZ-80B

COMMAND UP



このプログラムはSB-1520のモニタ・コマンドを追加、改良し、マシン語の入力を簡単にするものです。SB-1520のモニタ・サブルーチンを使いプログラムを非常に短かくしてあります。追加コマンドにはチェック・サム、トランスファなど大変役に立つものが多く、また、MZ-80Bの特色であるAPSSを利用し、ロード、セーブもかなり便利になっています。

プログラムの入力方法

①まずMON[CR]とし、モニタ・モードにします。ついで*M[CR]を実行し、\$12ACをFF・FDに変えます。そしてJ[CR]でJ-ADR、\$12A0を実行し、再びMON[CR]でモニタ・モードにします。

②リスト1を*M[CR]により、\$FD1B番地より打ち込んでいきます(途中でテープにSAVEし、次の②、続きを入力する場合も、必ず①を実行してからLOADし、そして、続きを入力していただくさい)。

③全部打ち込んだら、必ずチェック・サムを行なってください。異常がなければ*S[CR]、S-ADR.\$FD1B、E-ADR.\$FFFF、J-ADR.\$00B1とし、テープに取っておきます。

④そしてリスト2を*M[CR]により、\$C9A8番地より打ち込んでJ[CR]により、J-ADR.\$FE9Fでトランスファが使えのリスト1を\$C9F0番地からに移します(S-ADR.\$FD1B、E-ADR.\$FFFF、J-ADR.\$C9F0)。ついでリスト3を\$CCD6より打ち込んでください。

⑤以上ででき上がりなので、テープにセーブしておきます(*S[CR]、S-ADR.\$C9A8、E-ADR.\$CCE4、J-ADR.\$C9AB)。

コマンド表

コマンド	説明
*J(ジャンプ)	SB-1520と同じ方法。
*J(オート・ジャンプ)	プログラムをロード後のブレークの後にセーブのときに指定したJ-ADRに飛ばす。
*G(コール)	任意のアドレスをコールする。
*R(リターン)	BASICに返す(JP1300Hを行なう)。
*M(メモリ)	SB-1520と同じ方法。ただし以下の機能を持つ。 ①オート・リターン 2文字入力後、自動的に改行する。 ②・③改行後・(コマ)の入力により7つ前のアドレスに戻る。 ④+・+の入力により現在のアドレス~FC7Eを1

*O(ダンブ)	つ値にずらす。 ①-の入力により現在のアドレス~FC7Eを1つ前にずらす。 SB-1520と同じ方法。ただし、8バイト・チェック・サムが付いている。
*O(チェック・サム)	128バイト・チェック・サムを任意にS-ADR~E-ADRまでを表示する。
*C(クリア)	S-ADR~E-ADRまでを任意のDATAでぬりつぶす。
*F(ファンクション・サム)	ファンクション・キーを初期の状態に戻す。
*T(トランスファ)	S-ADR~E-ADRをJ-ADRからにブロック転送する。テン・キー、ファンクション・キーを①のように改連する。
*k(キー-OFF)	テン・キー、ファンクション・キーを元に戻す。
*S(セーブ)	SB-1520と同じ。ただし、セーブした後自動的にV-ERIFYをとる。
*s(コピー)	テープからのロード終了後、自動的にEJECTを行ない、テープを入れ換えると自動的に前のテープと同じS-ADR、E-ADR、J-ADR、FILENAMEでセーブされる。
*L(ロード)	SB-1520と同じ。
*I(返曲ロード)	数値を入力することにより飛び越し返曲し、ロード注: Lの小文字で数値を行なう。

図1 ファンクション・キーおよび16進キー

SHIFTで	↓	↑	→	←	H	C	■	□	○	1
ノーマルで	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
	7	8	9	+						
	4	5	6	=	E					
	1	2	3	-	N					
	0	00	.	T						

SB-5520との合成方法

COMMAND UPはそのまま使っても良いのですが、SB-5520と合成してIPIからロードできるテープにするとともに便利になります。方法は、

①01F5 E9~D7としておき、COMMAND UPをロードします。

②02H9から21、00、80と打ち込み、SB 5520をロードします。

③ アドレス	変更値	8119	BC	8666	31
80AF	D6	811A	FD	92AC	FD
80B0	4C	8134	E3	93E8	31
80D3	1B	8135	FF	CCD8	49
80D4	FD	813C	F0		
80D5	06	813D	FF		
80D6	14	814C	C9		
810E	FD	81BB	97		
810F	F1	81BC	FD		

と書き直す。

① *S コマンドで、

FILE NAME: BASIC SB 5521

S ADR. \$ 8000

E-ADR. \$ CCE4

J ADR. \$ 12A0

としてセーブしておきます。

使用上の注意

マシン語、BASICともにプログラム実行時には必ず、キーOFF(*k)を行なっておくこと、でないと暴走することがあります。

おまけ

① *D (ダンフ) コマンドのときの8バイト・チェック・サムを16バイト・チェック・サム (ただし、ヨコだけ) にする方法があります。方法は以下のとおりです。\$012Fを08・10にする。*G[CR]、J-ADR. \$0C7C[CR]、

元に戻す方法

\$012Eを10・08にする。

*G[CR]、J-ADR. \$0CEE[CR]

② *F (ファンクション復帰) をBASICで行なうことができ

ます。ダイレクト命令で、

POKE\$ 4985, 201:USR(\$ 4976):POKE\$ 4985, 195

③ POKE\$ 1140, 175:POKE\$ 1141, 195:POKE\$ 1142,

1:POKE\$ 1143,9とプログラムの初めの方で打っておき、

あとはGET文の前にUSR(\$ 1140)としておくとGET文

にだけオート・リビートが行なわれます。

④ このプログラムには*A、*a、*B、*bが予備としてとってあるので、FD1C、FD1F、FD22、FD25の各2バイトの値により4つのコマンドが追加できます。また、そこにプログラムのスタート・アドレスを入れておけばゲーム・スタート用のコマンドとして使えます。

⑤ COMMAND UPのコピーの方法はCOMMAND UPの*sコマンドを使うか、SB-1520の*Sコマンドである場合は、01F5をE9→D7としてから*Sコマンドを使ってください。

最後に

制作にあたって、ご協力いただいた三重電子機器販売KKの古井さんと水谷君には謝上を借りて、お礼申し上げます。

リスト 1

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
FD00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
FD10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
FD20	00	01	B1	00	62	B1	00	00	4A	02	40	F9	00	44	C2	FF	107
FD30	4C	6B	01	53	4E	01	54	9F	46	57	FD	4B	31	FE	6B	12A	
FD40	3E	FE	52	00	13	73	2B	FF	6A	F5	FF	43	60	FD	47	F9	179
FD50	FF	6C	E3	FE	64	5A	FF	21	B1	00	22	86	49	C3	76	49	14E
FD60	CD	7A	05	CD	B3	05	E5	11	BF	FD	CD	B6	05	11	93	10	15F
FD70	CD	C5	05	11	9A	10	CD	23	06	3B	E1	ED	5B	2B	00	1C0	
FD80	87	ED	52	EB	D5	C1	E5	D1	13	77	ED	B0	C3	B1	00	20	1EB
FD90	44	51	54	41	2E	24	0D	CD	B2	02	21	B4	FD	22	ED	04	1AF
FDAA	CD	B1	04	21	DA	04	22	ED	04	3E	12	32	D0	11	CD	CE	192
FDB0	04	C3	1A	02	3E	10	32	D0	11	C3	DA	04	FE	2E	2B	0C	145
FDCC	FE	2B	2B	0D	FE	2B	2B	18	CD	23	06	C9	D1	2B	2B	18	1C7
FDD0	1D	E5	54	50	13	01	7E	FC	CD	B7	FE	E1	D1	2B	2B	18	1C7
FDE0	E5	54	50	18	01	7F	FC	CD	B7	FE	E1	D1	2B	2B	2B	18	1C7
FDF0	00	E5	21	15	FE	22	39	07	E1	CD	AA	06	E5	21	40	0B	121
Sum	EF	0C	AF	1B	6F	3C	4E	80	D5	91	01	B2	D7	55	A5	9B	191

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
FE00	22	39	07	21	DF	06	36	CD	D1	0F	09	22	E0	06	E1	1A	1A7
FE10	FE	0B	2B	35	C9	E5	F5	3A	D1	11	FE	09	CC	25	FE	F1	10C
FE20	E1	CD	40	0B	C9	21	DF	06	36	3E	21	00	00	22	E0	06	16F
FE30	C9	FE	C1	32	B3	09	21	4C	FE	22	B4	09	18	0B	3E	79	10C
FE40	32	B3	09	21	FE	02	22	B4	09	C3	B1	00	79	FE	02	D2	1AD
FE50	B8	09	21	00	11	78	FE	04	C2	00	09	78	FE	06	3B	1A	193
FE60	FE	06	2B	0A	07	FE	2B	0A	01	7E	FE	C3	CF	09	3E	1D	100
FE70	1B	02	3E	1E	CD	29	0C	3C	CD	50	0C	C1	C1	C1	C1	21	102
FE80	D1	11	34	C3	AE	06	93	7F	3B	39	1A	2E	2B	2D	30	41	121

リスト 2

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
C9A0	00	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	12F
C9B0	00	21	06	14	22	B5	00	21	F1	FD	22	3E	01	21	BC	FD	14C
C9C0	22	19	01	21	E3	FF	22	74	01	21	F0	FD	22	3E	01	2E	14*
C9D0	22	19	01	21	E3	FF	22	74	01	21	F0	FD	22	3E	01	2E	14*
C9E0	FD	22	AC	12	7E	71	32	EB	17	51	00	FC	CD	06	CC	00	11B
Sum	EB	9F	FF	4B	64	9C	51	5F	0C	50	50	5B	75	96	B1	4C	1FE

リスト 3

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
CCD0	00	00	00	00	00	00	00	21	F0	C9	11	1B	FD	01	E5	02	ED	10B
CCD0	B0	C3	76	49	00	00	00	00	00	00	00	00	00	00	00	00	00	132
Sum	B0	C3	76	49	00	00	00	21	F0	C9	11	1B	FD	01	E5	02	ED	10A

①・②の機能が正常に作動するかを試してください。ダメならばSEC00-SECFE, SFF08-SFF7Bの間をチェックしてください。OKでしたらチェック・サム機能を利用してSED00-SFEFF間をチェックしてください。すべてOKでしたらS ADR-SEC00, E ADR-SFF80, J ADR-SEC10でテープにセーブします。そして各機能をチェックしてみてください。

BASIC SB-5520と共に IPLからロードする方法

BASICからLIMITSEBFFを実行しさらにモニタからJ ADR無指定の機械語プログラムをロードします (J ADR, SEC10を実行してはいけません)。

BASICプログラム・リストを入力してRUNすると“Command?”と聞いてきます。LM(Load mastertape の略)を入力すると“Set master tape Sure?(Y/N)”と出るのでSB-5520のテープを入れて[Y]を入力するとSB-5520のロード (S8000以後にロードされる) が始まります。ロード後、Tコマンド (Transfer) を実行しS ADR-SEC00, E ADR-SFF80, Destination-SCA00を入力すると先に入力したユーティリティ・プログラムがSCA00-1に移されます。その後いったんモニタに戻しS80AF番地からの1バイトを00,4Aに変更します。またBASICに戻りRUNして今度はBUコマンド(BACK UPの略)を実行します。“APPEND?”と聞いてきたとき[Y]を入力しNew end ADR-SDD80としFile name: Basic SB-5520 (省略しても可)で新しいテープにセーブしてください。

なおこのBASICプログラムは各種プログラム(機械語だけでなくBASICやPASCALも含む)のBack upに使用できます。

プログラムの説明

フローチャートは、書くとかなりやヤコシことになるので主なものアルゴリズムのポイントだけ書いておきます。
[Renumber]

S515C番地から格納されているBASICテキストは一行ごとに次の構造をとっています(どのBASICでも同じようですが)。

(次の行の先頭アドレスを示すポインタ・サイズ)・(1バイト表現の行番号)・(テキスト)

A B C

そこでまずAの部分に新行番号を書き込んで、新、旧行番号の対照がA, Bを見れば分かるようにしておきます。次にテキストの中から次の命令に対する中間コードを探します (“ ”内は読み飛ばします)。

GOTO, GOSUB THEN, RUN, RESTORE
THEN GOTO, THEN GOSUB, THEN RUN, THEN RESTORE

以上の中間コードが見つかったらその後にある文字列を2バイト表現の行番号に直します。それをA, Bの対照により新番号のASCIIコード列に直しテキストの該当部分に入れ替えます。すべてをリナンバーした後変数部分のクリアとポインタの書き換えを行います。

実際は行番号のオーバーフローチェックや、何かの目印のつもりでGOTO 100%などとなっていたら書き替えないなどの各種チェックを行ないながら実行しています。

[Delete]

これは指定行の部分を取り取り後でポインタを書き替えています。

[Search]

いったんテキストの一行ごとにBASIC内部のリスト・ルーチンの一部を利用してASCIIコード列に変換しその中から指定された文字列と同じものを探し、あればモニタ内のルーチンを使って表示しています。

[Arrange]

これは単に指定されたプログラムの部分について入れ替えを行なっているだけです。

[Append]

モニタのロード・ルーチンを利用してプログラムを現在のテキスト・エリアの後にひっ付けています。

[Variable]

変数テーブルに書かれているアドレスの内容を調べ、変数が存在すればそれを画面に表示しています。一部、BASICのプリント・ルーチンを使っています。

[Trace]

BASIC内部のS1C2D番地から分枝させて[CR]の入力ごとにリスト表示 (キャラクター画面の内容とリストを入れ替える) と実行を繰り返しています。

[Copy]

プリンタへの出力ルーチンはCALL 4755H (エスケープ・コードの出力はCALL 4753Hです) を利用しています。特に変わったことはしていませんが、VRAMのアドレスを\$5000-\$7FFFに切り替えるときの注意事項を書いておきます。通常VRAMへの切り替えと元に戻すときは次のサブルーチン呼びびます。

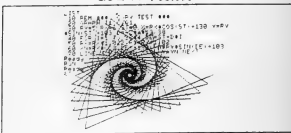
VRAM ON	VRAM OFF
V ON: DI IN A, (0E8H) OR C0H OUT(0E8H), A RET	V OFF: IN A, (0E8H) AND 3FH OUT(0E8H), A EI RET

しかし、ここで注意しなければならないのは現在のスタック領域がVRAMと重なる可能性があることです。すなわちLIMIT \$7FFFなどとしているスタックがVRAM内部になってしまいます。これを防止するためにスタック領域を一時\$5000-\$7FFF以外の場所に設定する必要があります。具体的には先に書いたサブルーチンを次のように変更すれば良いと思います。

V ON: DI	V OFF: POP HL
POP HL IN A, (0E8H) OR C0H OUT(0E8H), A LD (nn), SP LD SP, STACK JP (HL)	IN A, (0E8H) AND 3FH OUT(0E8H), A LD SP, (nn) EI JP (HL)

ここでSTACKは新しく設定したスタック領域、nnはSPの値の一時退避場所です (このプログラムではSTACK = FF00Hです)。

&Cコマンド実行例



実行例

LIST

```

10 REM *** Test program ***
20 REM
30 RESTORE 60:GOTO 60:GOSUB 60
40 ON A GOTO 50,60,70
50 IF A=A THEN 70
60 ON A GOTO 40,50,70
70 IF A=A THEN RUN 10
Ready

```

```

&AR 20-40,60
Ready

```

LIST

```

10 REM *** Test program ***
50 IF A=A THEN 70
60 ON A GOTO 40,50,70
20 REM
30 RESTORE 60:GOTO 60:GOSUB 60
40 ON A GOTO 50,60,70
70 IF A=A THEN RUN 10
Ready

```

FILE

```

NAME:
LOADING TEST
Ready

```

LIST

```

10 REM *** Test program ***
50 IF A=A THEN 70
60 ON A GOTO 40,50,70
20 REM
30 RESTORE 60:GOTO 60:GOSUB 60
40 ON A GOTO 50,60,70
70 IF A=A THEN RUN 10
10 REM *** Test program ***
20 REM
30 RESTORE 60:GOTO 60:GOSUB 60
40 ON A GOTO 50,60,70
50 IF A=A THEN 70
60 ON A GOTO 40,50,70
70 IF A=A THEN RUN 10
Ready

```

```

&R 1,,1
Ready

```

LIST

```

1 REM *** Test program ***
2 IF A=A THEN 7
3 ON A GOTO 6,2,7
4 REM
5 RESTORE 3:GOTO 3:GOSUB 3
6 ON A GOTO 2,3,7
7 IF A=A THEN RUN 1
8 REM *** Test program ***
9 REM
10 RESTORE 3:GOTO 3:GOSUB 3
11 ON A GOTO 2,3,7
12 IF A=A THEN 7
13 ON A GOTO 6,2,7
14 IF A=A THEN RUN 1
Ready

```

```

&D 10-
Ready

```

LIST

```

1 REM *** Test program ***
2 IF A=A THEN 7
3 ON A GOTO 6,2,7
4 REM
5 RESTORE 3:GOTO 3:GOSUB 3
6 ON A GOTO 2,3,7
7 IF A=A THEN RUN 1
8 REM *** Test program ***
9 REM
Ready

```

LIST

```

10 REM *** TEST ***
20 DIM A(5),BC(3,2),DE$(7),EF$(10,20)
30 G=1.2345:H$="ABCDE"
Ready

```

```

RUN
Ready

```

```

&S,"ABC
30 G=1.2345:H$="ABCDE"
Ready

```

```

&V

```

```

A (5)
BC(3,2)
DE$(7)
EF$(10,20)
G = 1.2345
H $="ABCDE"
Ready

```

SB-5520 ユーティリティ・プログラム

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EC00 21 10 4A 11 10 EC 01 70 13 ED B0 C3 10 EC 00 00 16B
EC10 21 B0 EC 11 F0 06 01 20 00 ED B0 2E DF 11 2F 01 +D0
EC20 0E 21 ED B0 21 31 FF 22 0E 06 21 6E FF 22 28 07 104
EC30 21 1F 1E 22 24 0E 21 93 7F 22 21 0E 21 AC EC 06 195
EC40 07 7E 23 5E 23 56 23 12 10 F7 1B 15 00 15 00 FF 1FC
EC50 2C 07 1E 0D 09 1D BF 09 1D 3C 08 1D E3 06 1E 2E +CE
EC60 07 3E A6 32 3F 16 21 20 FC 22 8F 1C 3E ED 32 AC 1B5
EC70 12 C3 76 49 42 00 00 00 00 00 00 00 00 00 00 194
EC80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
EC90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
ECA0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
ECB0 38 2C FE 07 30 2B 4F 3A 0D 11 CB 77 79 CA 1E 07 105
ECC0 F1 CD 4E FF C3 32 07 00 F1 CD 2A 09 FE 01 3B 04 14F
ECD0 CD 7A 05 CD 83 05 EB 2A 2B 00 CD BC 05 06 0E 10B
ECE0 00 CD C4 08 7E F5 81 4F F1 CD DD 05 05 AF ED 32 14F
ECF0 E1 CA 2D FF CD 62 05 CA 72 FF C3 08 FF 00 0E 52 +F5
-----
Sun 94 90 F4 B4 71 70 BC FD 9A 01 AF 04 90 F3 1C 54 1A7

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
ED00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF FF FF 1F8
ED10 AA 55 AA 55 AA 55 AA 55 00 00 00 00 00 00 00 1FC
ED20 08 08 08 08 08 08 08 08 08 24 24 24 00 00 00 194
ED30 24 24 7E 24 7E 24 24 24 08 1E 2B 1C 0A 3C 08 16B
ED40 00 62 64 08 10 26 46 00 30 4B 4B 30 4A 4A 3A 102
ED50 04 08 10 00 00 00 00 00 00 04 08 10 10 08 04 164
ED60 20 10 08 08 10 20 00 08 2A 1C 3E 1C 2A 08 00 152
ED70 08 08 3E 3E 08 08 00 00 00 00 00 00 08 08 10 17E
ED80 00 00 00 7E 00 00 00 00 00 00 00 00 00 1B 18 00 1AE
ED90 00 02 04 08 10 20 40 3C 42 4E 5A 62 42 3C 00 17C
EDA0 08 1B 28 08 08 08 3E 3C 42 42 0C 30 40 7E 10 118
EDB0 3C 42 0C 3C 02 42 3C 00 04 0C 14 24 7E 04 04 10A
EDC0 7E 40 78 04 02 44 3B 1C 20 40 7C 42 42 3C 00 170
EDD0 7E 42 04 08 10 10 10 3C 42 42 3C 42 42 3C 00 18B
EDE0 3C 42 42 3E 02 04 3B 00 00 00 00 08 00 00 00 14C
EDF0 00 00 08 00 00 08 08 10 0E 1B 30 60 30 1B 0E 00 134
-----
Sun 76 23 AB E3 76 B1 7E 65 49 C5 D5 3B 43 FB B1 0F 11A

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EE00 00 00 7E 00 7E 00 00 00 70 1B 0C 06 0C 1B 70 00 12A
EE10 3C 42 02 0C 10 00 10 1C 22 4A 56 4C 20 1E 00 114
EE20 1B 24 7E 42 42 42 00 7C 22 22 3C 22 22 7C 00 17E
EE30 1C 22 40 40 40 22 1C 00 7B 24 22 22 22 24 7B 00 15A
EE40 7E 40 40 78 40 40 7E 7E 00 40 7B 40 40 40 40 19A
EE50 1C 22 40 4E 42 22 1C 00 42 42 42 7E 42 42 42 15A
EE60 1C 08 08 08 08 08 1C 00 0E 04 04 04 04 3B 00 1FA
EE70 42 44 4B 70 4B 44 42 00 40 40 40 40 40 40 7E 10A
EE80 42 66 5A 5A 42 42 42 00 42 62 5A 46 46 42 42 12C
EE90 1B 24 42 42 42 24 1B 00 7C 42 42 42 42 40 40 17A
EEA0 1B 24 42 42 4A 24 1A 00 7C 42 42 7C 4B 44 42 00 192
EEB0 3C 42 40 3C 02 42 3C 00 3E 08 08 08 08 08 00 1EB
EEC0 42 42 42 42 42 42 42 3C 00 42 42 42 42 24 1B 18 106
EED0 42 42 42 5A 5A 66 42 00 42 24 18 24 42 42 42 18A
EEF0 22 22 22 1C 08 08 08 7E 02 04 1B 20 40 7E 00 114
EEF0 3C 20 20 20 20 20 3C 00 00 10 10 10 08 04 02 00 19A
-----
Sun FB EC B6 FA 76 AE D8 00 0B FA CB A2 AB F0 60 00 1FA

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EF00 3C 04 04 04 04 04 3C 00 1B 12 00 00 00 00 00 1CA
EF10 FF 00 00 00 00 00 00 20 10 08 00 00 00 00 00 137
EF20 00 00 3C 04 3C 4A 3A 00 40 5C 42 42 42 42 5C 00 188
EF30 00 3C 42 40 42 3C 00 02 02 3A 46 42 46 3A 00 1B2
EF40 00 00 3C 42 7E 40 3C 00 0C 10 10 7C 10 10 10 150
EF50 00 00 3A 46 3A 02 40 40 5C 62 42 42 42 42 00 142
EF60 08 00 1B 08 08 08 1C 00 04 0C 04 04 44 3B 1EC

```

SB-5520 ユーティリティ・プログラム

```

EFT0 40 40 44 48 50 68 44 00 18 08 08 08 08 10 10 16
EFB0 00 76 49 49 49 49 00 00 5C 42 42 42 42 40 11E
EF90 00 3C 42 42 42 3C 00 00 5C 42 42 42 42 40 13A
EFA0 00 3A 46 3A 46 3A 02 02 00 5C 40 40 40 40 182
EFB0 00 3E 40 3C 02 7C 00 10 7C 10 10 12 0C 00 112
EFC0 00 42 42 42 46 3A 00 00 00 42 42 42 24 18 04 148
EFD0 00 41 49 49 49 36 00 00 00 42 24 18 24 00 136
EFE0 00 42 42 46 3A 02 3C 00 00 7E 3C 22 7E 40 17A
EFF0 08 10 10 10 10 08 00 18 18 18 18 18 18 00 118

```

Sum 8B 54 4D 20 BA 14 CD 7A FA EA EA 6A 76 06 7B 199

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F000 10 08 04 08 08 08 10 00 00 00 32 4C 08 00 10 10 1C
F010 00 7C 04 04 15 0E 00 00 28 28 28 28 28 28 1E
F020 08 08 08 08 2A 1C 08 00 00 1C 2A 08 08 08 00 1CC
F030 00 08 04 7E 04 08 00 00 00 20 7E 20 10 00 00 174
F040 08 1C 3E 7F 1C 3E 00 36 7F 7F 7F 3C 1C 08 00 1F4
F050 08 1C 3E 7F 3C 1C 08 00 1C 1C 68 7F 68 1C 00 1F4
F060 10 10 10 1F 1F 10 10 10 10 10 10 10 10 10 10 10E
F070 00 00 00 FF 28 28 28 28 28 28 FF 00 00 00 00 116
F080 28 28 28 FF FF 28 28 28 10 10 10 FF FF 10 10 10 14C
F090 28 28 28 FF 28 28 28 28 00 00 00 FF FF 00 00 00 115
F0A0 22 14 3E 08 08 08 08 00 80 80 80 80 80 80 1F4
F0B0 00 3C 7E 7E 7E 7E 3C 00 00 42 42 42 42 42 40 10A
F0C0 00 00 00 30 10 10 10 10 10 10 10 F0 00 00 00 150
F0D0 00 00 00 1F 10 10 10 10 10 10 10 10 10 00 00 1AE
F0E0 10 10 10 FF 10 10 10 10 10 10 10 10 10 10 10 10E
F0F0 00 00 00 FF 00 00 00 00 10 10 10 FF 00 00 00 00 12E

```

Sum 82 8C 0C 3B 62 80 5E 8B BF 3C D5 0A 42 93 83 57 10A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F100 00 00 00 FF 10 10 10 10 10 10 F0 10 10 10 10 19F
F110 10 10 10 1F 10 10 10 10 80 80 80 80 80 80 1E7
F120 00 00 00 00 50 70 70 00 1E 10 10 10 00 00 00 17E
F130 00 00 08 08 08 7B 08 80 80 80 8A 55 99 80 FF 166
F140 80 80 8D EF D8 BD 80 FF 00 3E 02 3E 02 04 00 00 11C
F150 00 00 3E 02 0C 08 10 00 00 00 04 08 18 08 00 18B
F160 00 00 08 3E 22 02 0C 00 00 00 00 3E 08 08 3E 00 102
F170 00 00 04 3E 0C 14 3A 00 00 00 00 3E 02 04 10 00 10A
F180 00 00 1C 04 04 3E 00 00 00 3C 04 3C 04 3C 00 10E
F190 00 00 00 2A 2A 02 0C 00 00 00 00 D8 E3 F7 F7 FF 10C
F1A0 3E 02 04 0C 08 08 10 00 02 04 08 18 28 08 00 1D4
F1B0 08 3E 22 02 04 08 00 00 3E 08 08 08 08 3E 00 134
F1C0 04 3E 04 14 00 00 00 00 00 12 12 12 12 24 00 148
F1D0 08 3E 3E 08 08 08 00 00 1E 12 02 02 04 18 00 19A
F1E0 10 1E 24 04 04 04 00 00 00 3E 02 02 02 02 3E 1EA
F1F0 14 1E 14 14 04 08 00 00 30 02 32 02 04 38 00 138

```

Sum 43 E5 A5 61 09 9D 8B 1F 97 C1 01 B1 3F FF 3D 0D 10

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F200 00 3E 02 04 08 14 22 00 10 3E 12 14 10 0E 00 124
F210 00 22 12 02 04 18 00 00 1E 12 24 06 04 18 00 1F0
F220 04 3B 08 08 08 08 10 00 10 2A 2A 02 04 08 00 12E
F230 1C 00 3E 08 08 08 08 10 00 10 10 10 10 10 10 10E
F240 08 08 3E 08 08 10 20 00 10 00 00 00 00 3E 00 1E8
F250 00 3E 02 14 04 14 20 00 08 3A 08 1C 2A 08 00 130
F260 04 04 04 04 04 08 10 00 00 08 04 22 22 22 00 1C0
F270 20 3E 20 20 20 30 00 00 3E 02 02 02 04 18 00 17C
F280 00 10 04 02 0C 14 00 00 00 10 1B 14 14 10 00 1F2
F290 00 3E 02 14 08 04 00 1C 00 10 10 10 3E 02 00 19A
F2A0 00 08 10 20 22 3E 02 00 00 02 14 08 14 20 00 1EE
F2B0 00 3E 10 3E 10 10 0E 00 10 10 3E 12 14 10 10 00 15E
F2C0 00 1C 04 04 04 3E 00 00 3E 02 3E 02 02 3E 00 12A
F2D0 1C 00 3E 02 2A 2A 2A 00 20 20 22 24 28 30 00 136
F2E0 08 28 28 2A 2A 2A 00 00 20 20 22 24 28 30 00 86
F2F0 00 3E 22 22 22 3E 00 00 3E 22 02 04 08 00 10E

```

Sum 68 F8 C2 50 E8 20 AC 00 64 62 18 9C FE 38 7E 00 154

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F300 00 30 00 02 02 04 3A 00 10 48 20 00 00 00 00 1E8
F310 70 50 70 00 00 00 00 00 81 D0 FB EF D7 BF 81 FF 1AE
F320 E7 D8 BD 81 BD 80 FF 83 D0 CD D3 DD 83 FF 172
F330 E3 DD BF BF BF DD E3 FF 87 D8 DD 00 D8 BF FF 116
F340 B1 BF BF 87 BF 81 BF 81 BF 81 BF 81 BF 81 BF 14A
F350 E3 DD BF B1 BD 80 DD 81 BF 81 BD 81 BD 80 BF 19A
F360 E3 FF F7 F7 F7 F7 E3 FF F1 F8 BF BF BF BF BF 1FA
F370 8D BB BF BF BF 8B 80 FF BF BF BF BF BF BF 1E6
F380 8D 95 A5 8D BD 80 FF BD 9D A5 85 8D BD BF 1C4
F390 E7 D8 BD 80 BD E7 FF 83 BD 85 A5 8D BF 184
F3A0 F7 E7 BF F7 F7 F7 C1 FF C3 8D BF F3 C3 BF 11F
F3B0 C3 BD F0 C3 FD BD C3 FF BF F3 BF F3 BF F3 BF 188
F3C0 B1 BF BF BF BF BF BF BF BF BF BF BF BF BF 180
F3D0 B1 BD BF BF EF EF FF C3 BD 80 C3 BD 80 C3 FF 138
F3E0 C3 BD 80 C1 FD BF C7 FF 83 BD 83 BF BF BF 17C
F3F0 E7 D8 BD 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 15E

```

Sum 48 82 4A 8C 84 8B 66 F2 73 53 4D BD 65 95 4D F1 19E

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F400 C3 BD 80 BF 8D C3 FF C1 F7 F7 F7 F7 F7 FF 108
F410 00 00 01 3E 54 14 00 21 33 14 31 20 09 22 19 19E
F420 9D FC 21 00 FA C3 AB FC FE 32 C2 33 14 22 9D FC 109
F430 3E 2D 32 61 4D 8A BF FE 01 10 07 09 2B 11 17 FC 19B
F440 ED 88 13 ED 53 09 FE 21 00 F5 18 D9 CD 9E FE C2 131
F450 32 14 2A BF 2E 28 22 8B FE 3C 32 32 31 20 15C

```

```

F460 EA FE D4 22 B2 00 21 B2 F4 22 EE 01 C3 CB 01 ED 7B 1D5
F470 BF FE CD AE FA 2A BF FE AF 77 23 77 CD 10 OF C3 182
F480 B3 FB E1 CD B9 FA 19 22 B9 FE 11 00 01 19 3A 40 1A0
F490 11 BF FE FD 21 33 14 CD EA FA 11 BF BF ED 53 EA 103
F4A0 ED 01 CD AE FA 2A BF FE 22 B7 FE C3 B3 FB 3E 102
F4B0 32 B1 00 21 0D 3D 22 B2 00 21 B2 02 22 EE 01 21 1BF
F4C0 A2 F4 22 56 11 2A BB FE 22 54 11 ED 5B 52 11 C9 1FD
F4D0 CD 03 F9 E1 CD 09 18 FE 50 3A 4C F4 FE 52 20 21 181
F4E0 11 BF FE FD 21 33 14 CD EA FA 11 BF BF ED 53 EA 103
F4F0 11 B9 FE CD EA FA FD 21 03 F5 8D BF CD 100

```

Sum 48 12 C1 F6 A8 FC BC B0 21 C9 FF 5F 08 B6 B1 8C 50 1F

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F500 FA DD E9 2A DB FE AF CD AB FB FD 22 BB FE D5 11F
F510 2A BF FE AF CD AB FB 2A FE 22 B9 FE D1 D5 ED 1E3
F520 52 3B DE AF 2A BD FE CD AB FB 2A FE 22 D0 FE 1C0
F530 D1 E5 ED 52 D1 3B 4A B9 FE ED 52 30 C3 2A BB 1A1
F540 FE E5 2C AC FE CD 7B FB ED 5B AE FE A7 ED 52 44 110
F550 4D 2A BF FE ED 42 22 B9 FE E5 C3 11 0E 4B D5 2A 149
F560 BB FE E5 ED B0 2A BD FE ED 52 4D EB D1 ED 184
F570 B0 E1 C1 ED D0 B1 E1 ED 52 20 C3 DD 21 02 C3 182
F580 BD FB 2A BF FE ED 53 BB FE CD B9 FE 22 B9 FE 1C6
F590 ED 53 BD 1E AB E1 CD 9E FE C2 33 14 3E 04 32 182
F5A0 BF 3E 0A 32 B0 F6 AF 32 D9 F6 2A BB AF 32 D4 197
F5B0 FA 32 B0 F6 2A AF 32 D9 F6 2A BB AF 32 D4 197
F5C0 B0 F6 3E 32 D9 F6 2A BF 2A BF 2A BF 2A BF 32 D4 197
F5D0 2A BC 4F CD 7C F6 2A BA AF CD 1F F6 2A 90 4F AE 1E1
F5E0 BE CA F6 12 11 14 A8 D5 CD 6B F6 3E 24 15 3E 1C8
F5F0 3D 12 13 3E 22 12 13 0E 00 12 D1 CD B9 08 4E AB 1A4

```

Sum 01 75 9E 3C 21 C9 92 17 37 92 B6 23 67 A2 99 4D 174

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F600 CD 2D 0C AF BB 28 03 13 1A CD 50 0C 23 10 FB 3E 158
F610 22 CD 50 0C AF BB 28 03 13 1A CD 50 0C 23 10 FB 3E 158
F620 52 ED 43 BD FE AF BE CB 01 51 F6 ED 43 00 C1 11 18A
F630 14 AB 3E 22 12 13 E5 CD 6B F6 21 F1 F6 0E 03 ED 170
F640 B0 E1 CD 6B F6 22 B9 FE 3E 00 12 21 14 4B C3 49 8B1
F650 3A ED 7B BF FE 2A BD FE E5 21 0A 18 22 00 1C DD 177
F660 DF FB 2A BF FE 11 05 00 19 18 A1 01 ED 00 80 159
F670 C9 32 22 38 AF BE CB CD 6A 18 FB 7E 23 8A 28 173
F680 CB CD 6A 18 FB 11 14 AB 01 02 00 ED B0 18 00 147
F690 3E 24 12 13 3E 28 12 13 7E 23 E5 32 84 F6 6E 26 111
F6A0 CD 02 92 FB CD AF FB E1 4E 23 E5 26 00 69 E5 18 18C
F6B0 00 3E 2C 12 13 CD 92 FB CD AF FB E1 3E 29 12 13 1C5
F6C0 3E 2C 12 13 CD 92 FB CD AF FB E1 3E 29 12 13 1C5
F6D0 B4 FE CA 01 23 CD EF FA ED 0B 3E 05 CD EF F6 D1 12C
F6E0 19 C9 EB E1 23 3E ED 0B 81 1B 7A B3 20 F6 C9 EB 12C
F6F0 21 00 00 06 09 30 01 19 10 01 C9 29 B7 18 F6 00 112

```

Sum 8C 01 BA C7 ED 40 8B C6 3B 31 9B 9F 2A CD B4 35 1A2

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F700 00 00 00 00 00 00 00 00 00 00 00 E1 CD 09 18 08 CD 1A4
F710 9E FE C2 33 14 04 BF FE 46 CA AF FC FE 46 0F 23 21 1E6
F720 E7 2E 22 1E 2C 21 B9 F7 22 1C 1C 3A F2 E1 C1 11E
F730 B2 18 2B ED ED 5B 9F FE E5 AF ED 52 E1 2B 44 22 155
F740 B9 FE CD 92 F7 C2 42 1C AF CD AB FB 2A CA FE CD 190
F750 F1 BF AF 03 AF 73 BF FE CD 44 FE 11 00 50 21 100
F760 00 4A CD BD 07 CD 50 FE CD 92 F7 08 CD 44 FE 11 171
F770 00 4A 21 00 50 CD BD 0F CD 50 FE CD 9A B6 55 D3 104
F780 00 50 C3 00 50 CD BD 0F CD 50 FE CD 9A B6 55 D3 104
F790 00 14 C7 32 08 FE 00 CB FE 08 00 02 02 C9 0E 13 16A
F7A0 CD B1 F7 C8 57 20 EB 0E 1B CD B1 F7 C8 57 20 E2 164
F7B0 C9 08 EB E6 F0 B1 D3 EB DB EA DB EA C9 4E 1A 77 100
F7C0 79 12 23 13 0E 00 20 F5 C9 CD 03 F9 E1 11 BB FE 11E
F7D0 21 E1 CF CD EA FA FD 21 F4 F7 11 B9 FE CD E4 12E
F7E0 FE 32 33 14 06 00 00 00 CD FB C7 7A B3 28 F1 11C
F7F0 ED 53 B9 FE DD 21 39 1A 2A B9 FE ED 5B BB FE D5 1F9

```

Sum CD A3 DF 9E AE B3 F1 2C 09 7F 09 BE 8B 4B 33 A7 167

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F800 E5 ED 52 38 CD D1 2A BD FE ED 52 20 09 D1 D5 2A 126
F810 5E 51 ED 52 38 CD E1 AF CD AB FB FD E5 2A 126
F820 B9 FE AF CD AB FB 2A CA FE D1 E5 ED 52 EB 2A B7 10E
F830 FE ED 52 3E 22 1E 2A AF 1C ED 52 44 EB D1 17E
F840 ED B0 DD 21 49 FB C3 B0 FB E5 2A B7 ED 5B AE 1A0
F850 4F ED 52 E9 31 4E AF 04 2A E1 19 E5 E1 10 FA C3 1E0
F860 02 FC E1 DD 21 C9 FB CD 09 18 FE 2C C2 33 14 0E 1C
F870 00 23 E5 FE 18 28 29 03 2C 79 FE 29 28 05 18 10E
F880 F2 E1 AF 89 CA 33 14 7B 32 AF FE 06 00 11 2B F9 104
F890 FE ED 52 3E 22 1E 2A AF 1C ED 52 44 EB D1 17E
F8A0 21 0E 4A 3E 20 ED B1 EA AF 7E 47 21 2B 9A 1A 1E2
F8B0 13 FE 0D 28 0F BE 20 F7 D5 10 14 11 00 4A CD B6 10E
F8C0 05 CD DF CD 62 05 20 CF CD 2E 0A C3 F6 12 23 18F
F8D0 1A 13 FE 0D 28 06 BE 28 0D 11 8C CD 11 E5 CD 17C
F8E0 32 08 FE DD CD 32 08 20 C8 FE 08 2B DA 18 12B
F8F0 F4 ED 5B AE FE A7 ED 52 44 EB D1 17E 0E 4B ED 151

```

Sum 50 57 92 2E 9C BD 04 40 62 A4 DD 32 1A 5A EF 95 121

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
F900 C3 80 15 DD 21 0D F9 AF 67 6F C3 4B FB D1 2A B0 195
F910 FE 23 25 23 E5 22 36 ED 22 B9 FE 22 D0 FE 21 E4 F7 18B
F920 22 E6 FA DD 21 33 14 C9 00 00 00 00 00 00 00 00 100
F930 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
F940 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

SB-5520 ユーティリティ・プログラム

```

F950 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
F960 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
F970 28 F9 05 E5 01 0A 00 C5 B0 21 96 00 C1 D1 C5 16 F0
F980 E5 ED B0 D1 C1 E1 ED B0 21 6A 00 11 A2 00 46 1A 14A
F990 77 72 12 03 F6 12 00 F2 00 C9 F9 21 0A 00 22 89 1A9
F9A0 FE 22 B0 FE 21 FB FA 22 6A FA 11 B9 FE CD E4 14A
F9B0 FA 11 B8 FE CD E4 FA 11 B0 FE CD E4 FA C3 13 1F0
F9C0 2A B8 FE DD 21 B9 FB CE 01 CD 4B FB 11 5C 51 2A 1CF
F9D0 FE FE E5 ED 32 28 10 2A B0 FE 23 23 25 56 2A 127
F9E0 B9 FE E5 ED 32 30 1E DD 21 08 FA E1 22 CE FE 1A9
F9F0 78 FB 2A AE FE ED 5B B9 FE 23 23 72 2A B0 FE 19 151
Sum 6B CC 39 F2 CE 4E 2A DB C0 47 78 4A 41 C7 F7 B2 1CD

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FA00 22 B9 FE 30 EA C3 B9 FB 0E 20 21 AB FA 21 5D 51 1BD
FA10 06 00 0E 00 16 22 23 23 23 7E B0 28 17 89 28 0B 1A3
FA20 BA 20 F5 23 7E BA 28 F0 B9 20 ED 23 7E 23 86 CA 14C
FA30 CA FA 1B E2 CD 09 FC 28 10 38 DF 32 B9 FE CD 09 13C
FA40 1B B8 20 09 CD 09 FC 20 D1 32 89 FE 23 2B 06 3A 1E3
FA50 CD FE FA CA AB FA AB E4 CD 4B FB 5D 3E 00 56 38 FE
FA60 01 EB DD 92 FB 01 E4 FE 96 28 39 F5 30 01 05 4F 1BA
FA70 2A B7 FE E5 09 22 B7 FE 2A B5 FE 54 50 09 22 142
FAB0 B5 FE E1 A7 ED 52 2A AE 0A F1 39 00 28 54 50 137
FA90 09 EB ED 4B AE FE ED B8 18 0A 62 6B 09 EB ED 4B 194
FAA0 AA FE ED B0 ED 5B FE CD 47 FB ED 7B BF FE 3A 13B
FAB0 B9 FE FE 2A B5 FE 7E 28 05 FE 2C CA 4E FA 0E 113
FAC0 00 06 B9 CA 28 FB C3 18 FA DD 21 83 FB 21 5C 139
FAE0 51 22 AC FE CD 7B FB FA FE E5 5D 13 13 ED A0 19A
FAF0 AD A0 18 F0 D5 CD FB FA E1 2B 03 73 23 2D B5 11C
FAF0 FE 7E DD C0 01 ED F9 0E 00 CD FE FA C9 0E 2C 1DC
Sum 24 D6 F9 6E 00 81 2A A4 B7 39 9D B8 8B F0 98 FD 10C

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FB00 CD 09 18 22 B2 FE 1E DD 16 00 7E B0 28 0A B9 28 14A
FB10 07 B8 28 04 14 23 18 2A 32 84 FE 22 B5 FE 2A 18C
FB20 B2 FE CD 26 FB C9 CE C3 32 13 11 3B FB ED C3 16B
FB30 34 14 CD C1 18 2A B5 FE B8 02 DD 05 3E 3E 32 123
FB40 33 14 21 01 01 22 34 1A 7A B3 C9 47 22 AB FE 21 55A
FB50 5C 51 22 AC FE 08 CD 75 FB FD 2A FE FE 5D 02 1EE
FB60 FD 56 03 2A AB FE 07 ED 52 CB 08 2B ED 73 00 15C
FB70 FD 72 01 18 0E 2A AE FE 22 B0 FE 2A AE FE 22 AE 1B2
FB80 FE 7E 2A 28 B9 23 23 3E 0D 47 ED B1 22 AC 199
FB90 FE C9 D5 CD 0E 19 01 01 01 21 84 4B 3E 00 ED 1B9
FBA0 B1 79 2F 32 A7 FE C9 3A 7F 4E 0F 06 21 84 4B 14D
FBB0 ED B0 C9 DD 21 CF FB 18 04 DD 21 39 14 21 5C 163
FBC0 22 AC FE CD 7B FB FE 2A 7E 23 18 FA 0E 106
FBD0 B7 FE E5 11 46 00 19 ED 73 AB FE 31 96 4F 0E 2A 150
FBE0 E5 2B 10 FB ED 7B AB FE E1 5D 54 28 0E 4B ED 154
FBF0 B0 2A 0A 4A ED 5B 2A B1 00 0E B8 ED 42 3F 76 CD 1E9
Sum 4B 72 29 C6 05 3D 1A 56 64 65 1A DF 48 68 E5 154

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FC00 B8 18 2F 32 EB 14 C3 F6 12 7E FE B6 CB FE 9C 105
FC10 CB FE 8C 28 09 D8 FE B8 C0 01 AF 3F C9 47 05 187
FC20 ED 73 BF FE CD 0A 18 0E FF E5 54 5D 28 23 0C 7E 187
FC30 FE 0D 28 04 FE 3A 20 05 22 F5 51 2A 4E 4F 22 87 1EB
FC40 FE 2A 5D 51 22 B8 FE DD 21 33 14 AF B9 2B 30 1A 0E
FC50 FE 43 CA BE FC DD B2 1B C2 64 1A 1E 4C 28 2F 126
FC60 18 0E 4B CA 65 F9 2A 3C 51 7C B5 28 1E 1A FE 1EF
FC70 56 CA 96 F5 FE 44 CA C9 F7 FE 41 CA D0 FA FE 54 196

```

```

FC80 CA OA F7 D6 52 CA 97 F9 30 CA 62 F8 C3 33 14 E1 199
FC90 CD 09 18 F5 CD 9E FE 20 F3 F1 FE 33 C2 18 F4 21 170
FCA0 33 14 22 BF 1C 21 00 F0 7C 32 AC 12 22 08 4A 21 126
FCB0 62 05 22 2E 1C 21 00 22 4B 1C C3 F6 12 E1 CD 1C7
FCC0 9E FE 20 CB 11 FC 18 D5 3A B6 45 F6 0F 3A B6 193
FCD0 45 DD 21 F3 4B 47 ED 43 AB FE ED 73 BF FE DB EB 17D
FCE0 CB 6F 3E CD 21 E2 FD 11 1F FE 01 DA FD 2B 0F 02 184
FCF0 3E 50 77 3E 01 32 9A FD 12 21 4F 50 18 0D 12 3E 154
Sum ED 96 27 FB 77 62 30 85 12 49 B3 OF 66 70 B9 D3 1E2

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FD00 28 07 AF C3 9A FC 3D 02 21 27 50 22 B2 FE 21 27 107
FD10 60 22 B5 FE CD 26 47 CD 74 46 CD 53 47 CD 35 47 1C6
FD20 32 B0 FE 21 26 FE 77 DB 0E CB 67 20 03 3E 27 77 190
FD30 AF 32 13 FE 3C 21 CF FD 11 D4 FD 17 12 3A AB FE 166
FD40 4F 05 CD 49 3E CD 28 01 77 CB 51 28 01 12 CB 59 21A9
FD50 05 3E 0B 32 13 FE 3E 19 32 AA FE AF 21 B0 FE BE 1FE
FD60 01 B9 FE 11 B2 FD 2B 0A 35 12 32 9C FD 3E 03 02 197
FD70 18 0A 34 02 3E 04 12 32 9C FD 2A B5 FE 22 AE FE 120
FD80 E5 18 00 28 22 B5 FE E1 11 FF 5F 07 ED 52 20 45 157
FD90 CD 00 C9 2A B5 FE 22 AC FE 00 18 00 2B 22 82 187
FDA0 FE CD 53 47 3E 18 CD 55 47 3E B8 CD 55 47 3E 1C2
FDB0 CD 55 47 4F 0E CD 28 55 47 10 FB 06 08 21 B7 FE 19E
FDC0 E5 AF 77 23 10 FC D1 CD 44 FE 3A 49 FE CB 87 CD 11A
FDD0 5D FE CD C7 CD 5D FE 22 AE FE CD 70 FE 2A AC FE 1FF
FDE0 05 09 09 09 22 AC FE 0B CD 50 FE 3A 49 FE D3 128
FDF0 F4 0B D6 01 3B 08 FE 06 3B 09 D6 1D 30 0A 01 7C 196

```

```
Sum 07 F1 F5 A7 17 93 51 13 D1 33 42 0E EC AB DE 04 141
```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FE00 FE 18 0F 01 C1 FE 18 03 01 0B ED 26 00 6F 29 1DD
FE10 29 09 18 00 06 0B D5 1A B6 12 13 23 10 F9 D1 CD 10C
FE20 7D FE 06 0B 5C 1A 2F 4F 06 02 CD 55 47 79 10 FA 1ED
FE30 C1 13 10 D0 F0 21 AA FE 3C 22 BB FD CD 62 05 3E 198
FE40 FD E1 DD 09 F3 E1 31 00 FF DB FE 06 CD 03 EB 9E 1C5
FE50 DB ED E6 3F D3 E8 E1 ED 7B BF FE FB E9 D5 D5 12A
FE60 FA 2A AE FE 06 0B C5 7E 01 00 0B 87 CB 19 10 FB 15A
FE70 1A B1 12 0E 2F 09 13 C1 10 EC F1 B1 C9 06 08 D5 15A
FE80 21 A7 FE C5 01 00 0A 1A 18 00 ED 67 7E 17 F5 CD 16B
FE90 11 F1 CB 11 10 F7 79 12 13 C1 10 E7 D1 C9 CD 09 1AB
FEA0 1B FE 0B 08 FE 3A C9 00 00 00 00 00 00 00 00 00 1EC
FEB0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
FEC0 00 1C 14 14 77 22 14 08 00 00 00 00 14 22 77 14 1C 1F2
FED0 00 0C 0C 7A 41 7A 0C 0B 00 08 1B 2F 41 2F 18 0B 13C
FEE0 00 77 55 5D 41 5D 55 77 00 1E 21 4F 00 5F 21 1E 1FF
FEF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
Sum 95 07 0B B6 A9 CE BF 80 35 4C F3 A2 4D 1F D0 E8 14D

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
FF00 00 00 00 00 00 00 00 00 00 DB EA FE FD 2B FA 23 05 10A
FF10 C2 31 01 CD C4 0B 3E 3A CD C6 08 79 CD DD 05 C3 18B
FF20 2A 01 CD C4 0B 3E 3A CD C6 08 79 CD DD 05 C3 18B
FF30 00 CD 0F 0F 10 2B 11 32 6F FF 21 6C FF 36 50 1E7
FF40 21 6D FF 36 5A AB FF C9 0E 0C 21 6C FF 36 50 1E7
FF50 0D 2B 5E 3E FF BE 23 2B F5 21 6C FF AF BE 2B 0A 17A
FF60 35 3E 1C D9 23 BE 2B DB 35 18 F6 0D 50 05 25 F5 197
FF70 01 05 01 21 40 00 CD 22 0F 21 6C FF 00 00 00 00 1E0
Sum 50 D7 DE FB 31 19 23 29 AD 5B AD 5A 31 AF 4C DF 1B2

```

BACK UP用BASICプログラム

```

10 REM *****
20 REM *** Backup program ***
30 REM *****
40 PRINTCHR$(6);:LIMIT$=7E5F:CONSOLE:PRINT"***** Backup program *****"
50 IFPEEK($7E68)>23760SUB690
60 PRINTSPACE$F:USR($0F14):GOSUB700:PRINT"Command? ";:GOSUB740:IFA$=1THEN60
70 FORI=2101STEP-1A$=LEFT$(A$,1):RESTORE:FORJ=1T021:READB$:IFB$>A$THENNEXTJ,I
:GOTO60
80 DATA,BU,T
90 NGJG0T0190,300,100
100 GOSUB510:IFF=2THEN60
110 AD=AS:GOSUB580:IFF=2THEN60
120 AE=AS:GOSUB700:PRINT"Destination -$";:GOSUB770:GOSUB520:IFF=2THEN60
130 A=AS:AS=AD:AD=A:IFA$<ADTHEN160
140 IFAD<32768THEN180
150 FORI=ASTOAE:POKEAD,PEEK(I):AD=AD+1:NEXT:GOTO60
160 AD=AD+AE-AS:IFAD>65543THEN180
170 FORI=ATOASSTEP-1:POKEAD,PEEK(I):AD=AD-1:NEXT:GOTO60
180 PRINT"Illegal data":USR($0F14):GOTO60
190 PRINT"Set master-tape,":PRINT"Sure? (Y/N) ";:GOSUB740:IFA$="1"THEN60
200 AS=LEFT$(A$,1):IFA$<":PRINTNUSR($0F14):GOTO190
210 POKE$7EA2,$C3:POKE$7EA9,$22:USR($7E92):PRINT:IFPEEK($7E91)=1THENPRINT"$Brea
k":USR($0F14):GOTO60
220 PRINT:A=PEEK($1140):IFA$=160THENPRINT"PASCAL interpreter text file":GOTO612
230 ONAGOT0240,250,260,270,280
240 PRINT"Object file":GOTO290

```

BACK UP用BASICプログラム

```

250 PRINT"BASIC interpreter text file":GOTO290
260 PRINT"BASIC or PASCAL interpreter data file":GOTO290
270 PRINT"ASCII file":GOTO612
280 PRINT"Relocatable binary file"
290 GOSUB710:AS=AE+32767:GOSUB590:PRINT"Load address=";"$B000~";AS$:GOSUB730:GO
TO60
300 PRINT"Program is already loaded? (Y/N) ";:GOSUB740:IFA$="" THEN60
310 AS$=LEFT$(A$,1):IFA$="N" THENUSR($0F14):GOTO190
320 IFA$<"Y" THENUSR($0F14):GOTO300
330 PRINT"Append? (Y/N) ";:GOSUB740:IFA$="" THEN60
340 AS$=LEFT$(A$,1):IFA$="N" THEN400
350 IFA$<"Y" THENUSR($0F14):GOTO330
360 AS=PEEK($1152)+256*PEEK($1153):AE=AS:GOSUB590:B$=AS$:AS=AE+32767:GOSUB590:P
RINT"Load address=$B000~";AS$;" (" ;B$;" bytes)"
370 PRINT"Input new end address -$";:GOSUB770:GOSUB520:IFF=2THEN60
380 IFAS<32768THENPRINT"Illegal data error.":USR($0F14):GOTO60
390 AS=AS-32767:A=INT(AS/256):AS=AS-A*256:POKE$1152,AS:POKE$1153,A
400 A$="" :FORI=441704433:A=PEEK(I):IFA$>13THENA$=A$+CHR$(A):NEXT
410 PRINT"File name:";A$:PRINT"File name for change:";:GOSUB740:IFA=11THEN60
420 IFA$="" THEN440
430 AS$=LEFT$(A$,16)+CHR$(13):A=LEN(A$):J=1:FORI=441704416+A:POKEI,ASC(MID$(A$,
J,1)):J=J+1:NEXT
440 PRINT"Set new tape.":PRINT"Sure? (Y/N) ";:GOSUB740:IFA=11THEN60
450 IFA$<"Y" THEN440
460 USR($7E68):PRINT:IFPEEK($7E91)=1THENPRINT"*Break":GOTO60
470 PRINT"Save again? (Y/N) ";:GOSUB740:IF(A=11)+(A$=""N) THEN60
480 IFA$="" THEN440
490 GOTO470
500 POKE$7F90,AX:POKE$7F74,AX*4:POKE$7F1D,AX:POKE$7F2D,AX:RETURN
510 F=0:GOSUB700:PRINT"Start address -$";:GOSUB770
520 IFA$="" THENF=2:RETURN
530 AS$=RIGHT$("000"+LEFT$(AS$,4),4):AS=0:FORI=1T04:A$=MID$(AS$,I,1):GOSUB550:I
FF=2THENRETURN
540 AS=AS*16+A$:NEXT:RETURN
550 IF(A$>" /" )$(A$<" :") THENA=VAL(A$):RETURN
560 IF(A$>" 0")$(A$<" G") THENA=ASC(A$)-55:RETURN
570 F=2:RETURN
580 GOSUB700:PRINT"End address -$";:GOSUB770:GOSUB520:RETURN
590 AS$="" :A=INT(AS/4096):GOSUB610:AS$=AS$+A$:AS=AS-A*4096:A=INT(AS/256):GOSUB6
10:AS$=AS$+A$:AS=AS-A*256:A=INT(AS/16):GOSUB610:AS$=AS$+A$:A=AS-A*16
600 GOSUB610:AS$="" $"+AS$+A$:RETURN
610 IFA<10THENA$=STR$(A):RETURN
620 DNAGOTO,,,,,,630,640,650,660,670,680
630 A$="A":RETURN
640 A$="B":RETURN
650 A$="C":RETURN
660 A$="D":RETURN
670 A$="E":RETURN
680 A$="F":RETURN
690 RESTORE780:FORI=32360T032485:READA:POKEI,A:NEXT:RETURN
700 PRINTSPACE$(FD):CURSOR0,CSR:RETURN
710 AS=PEEK($1154)+PEEK($1155)*256:AD=AS:GOSUB590:PRINT"Real address=";AS$:AS=
PEEK($1152)+PEEK($1153)*256:AE=AS:GOSUB590:B$=AS$:AS=AE+AD-1:GOSUB590
720 PRINT"~";AS$;" (" ;B$;" bytes)":RETURN
730 AS=PEEK($1156)+PEEK($1157)*256:GOSUB590:PRINT"Exec. address=";AS$:RETURN
740 J=CSRH:POKE$06A2,39:USR($7E8A):A$="" :IFPEEK($7FAE)=11THENA=1:RETURN
750 FORI=32686+JT032725:A=PEEK(I):IFA$>13THENA$=A$+CHR$(A):NEXT
760 RETURN
770 GOSUB740:AS$=A$:RETURN
780 DATA237,115,14,127,62,195,50,177,0,33,175,126,34,178,0,205,81,2,243,22,2,23
7,75,82,17,33,0,128,205,90,2,195,175,126,17,174,127,205,164,6,201,0
790 DATA237,115,14,127,62,195,50,177,0,33,175,126,34,178,0,62,195,50,237,1,33,2
10,126,34,238,1,195,203,1,62,62,50,177,0,33,13,211,34,178,0,62,205,50,237,1
800 DATA33,178,2,34,238,1,62,0,48,1,60,50,145,126,42,14,127,249,201,42,84,17,22
9,33,0,128,34,84,17,205,178,2,225,34,84,17,195,175,126

```


MZ-80

K/C↔B

テープ・フォーマットの変換法

■大垣泰二

シャープから新発売のMZ-80Bは、カセットのデータ転送速度が2,000ボーになり、K/Cタイプ(1,200ボー)のテープを読み込んでくれます。

そこで「また、テープ変換プログラムを作ったかせこうノ」と思い、マニュアルに載っているモニタのリストを良く読んでみると……? なっ、なんと、たったの1バイトを書き換えるだけでK/Cタイプのテープをそのまま読み込んでしまうのです。

Bタイプのメイン・メモリはすべてRAMだし、モニタにはメモリ書き換えの機能があるので、この作業はいとも簡単に実行できます。つまり、私は儲けそこなってしまうのです。くやし涙にくれて、そのやり方を報告します(大げさー/)。

変更点について

0557H番地の41Hを66Hに書き換えれば良いのです。その上、さらに4バイトのデータを書き換えれば、K/Cタイプで読み込めるテープを書くことも可能になります(表1)。

いまところ、実験したのは機械語のオブジェクト・テープ、つまり、01タイプのテープですが、K/Cタイプの例からみてほかのテープも結局同じサブルーチンで

読み書きされるので、同じ方法が通用すると思います。シャープから出されている「K/C→B, BASICテキストコンバータ」もこの方法が使われているようです。

なぜなら、モニタ・リストのコメント欄には上記の値がチラリと書かれているのです。

ただ、注意して欲しいのは、K/Cタイプの機械語テープが読み込めるということがすぐに、K/Cタイプの機械語プログラムをBタイプがそのまま実行できない点です。モニタ・サブルーチンのアドレスやI/Oの番地が、まったく異なっているのですから、同等の動きのサブルーチン番地に書き換えなければなりません。

しかし、K/Cタイプのシステム・プログラムを使っていれば、モニタ・サブルーチンをラベルで定義できるのであまり苦にならないでしょう。

り前と言えば当然前ののですが、転送速度を大幅に上げたのですから、フォーマットがもっと変わっていても不思議ではないわけです。

この図1を見ながらテープリード・ルーチンの方も検討してみると、なんと変更した部分は単なる導入部で、何サイクル繰り返されようがまったく関係がないのです(ただし、インフォメーション・ブロックの前後にある「空白」は、飛び越しサーチのときに必要だと思います)。

つまり、テープ・リード・ルーチンから見ると、K/CタイプのテープとBタイプのテープはスピードが異なるだけなのです。そして、MZではそのスピードがソフトで決定できるので、前記のようなわずかな変更によって両タイプのテープ交換ができるのです。

なお、変更したデータはすべてタイ・ルーチンにセットされるカウンタの値です。

テープ・フォーマットの比較

報告は以上ですが、これだけではあまりにそっけないので、なぜこれだけで済むか、その理由についても少し書いておきます。図1を見てください。これはK/CタイプのモニタSP-1002とBタイプのモニタSB 1520を解析して書いた両タイプのテープ・フォーマットです。良く似ているでしょう。同メーカーの同じシリーズですから、当

表1 K/Cタイプ・テープ作成のための変更点

アドレス	現在の値	変更値
0 5 2 4	2A	3F
0 5 3 0	25	3A
0 5 3 F	5A	81
0 5 4 B	55	7C

図1 MZ-K/CタイプとBタイプのテープ・フォーマット比較

K/C タイプ	インフォメーション・ブロック										データ・ブロック	
	送サイクル 22,000回	送サイクル 40回	送サイクル 40回	最 大 128 バイト	インフォメーション 1回目	チェック サム 2バイト	最 大 128 バイト	送サイクル 128回	インフォメーション 2回目	チェック サム 2バイト	最 大 128 バイト	
B タイプ	空白 8秒	送サイクル 10,000回	送サイクル 40回	送サイクル 40回	最 大 128 バイト	インフォメーション 1回目	チェック サム 2バイト	最 大 128 バイト	送サイクル 256回	インフォメーション 2回目	チェック サム 2バイト	空白 8秒
K/Cタイプ	データ・ブロック										データ・ブロック	
	送サイクル 11,000回	送サイクル 20回	送サイクル 20回	最 大 128 バイト	データ1回目 Nバイト	チェック サム 2バイト	最 大 128 バイト	送サイクル 256回	データ2回目 Nバイト	チェック サム 2バイト	最 大 128 バイト	
Bタイプ	送サイクル 11,000回	送サイクル 20回	送サイクル 20回	最 大 128 バイト	データ1回目 Nバイト	チェック サム 2バイト	最 大 128 バイト	送サイクル 128回	データ2回目 Nバイト	チェック サム 2バイト	最 大 128 バイト	

PC→MZB CONVERTER

■FALCON・S

「81年7月号の「PC→MZ変換プログラム」を見て私は小踊りして喜びました。もしかしら、これを応用して「PC→MZ-80B用の変換プログラム」が作れるのではと思ったのです。80Bはまだ発表になったばかりでソフトがほとんどありません。その点、PC用のシステム・プログラムは充実しているので、コンパタがあると大変に役に立ちます。現在PC上で80B用のプログラムを開発して80Bへ移しています。ただし、マシン語のプログラムしか変換できません。

1. 操作法

PC-8001でテープにSAVEしてから、まず頭出しをしておきます。テープの音を聞くとわかると思いますが、最初に「ピー」という音の次に「ガー」という音が始まります。この「ピー」という音のする所でテープをストップして取り出して、MZ-80Bにセットします。

80BはMONコマンドでモニタにしておき、「J」コマンドを使って6000番地からスタートさせます。画面上にタイトルと「LOADING PC-TAPE」が表示され、後はもうやることはありません。終わるまで待つていけばいいのです。

リーダーテープの部分から始めたりするとエラーが出る場合があります。このときは「BAD:CR=MON」と表示されます。「CR」キーを押すとモニタに戻れるので最初からやり直してください。

読み込みが終わると「OK:CR=MON」と表示されます。「CR」キーでモニタに戻れます。

2. プログラムについて

プログラムの内容は「81年7月号の「PC→MZ変換プログラム」とほとんど同じです。80Bではモニタ・サブルーチンのアドレスや、テープからのデータ入力ポートが異なるのでその変更をしました。Hレベル・Lレベル判定のカウント値はよくわからなかったのです。実際に動かしながら「カット・アンド・トライ」で決めました。短いプログラムだし、アセンブル・リストが載っているので内容はわかると思います。

3. 使用感

完成してからいまでのところ、段々エラーなく動いています。80Bのモニタの内容を完全に読んでいないので、何となくこの辺だろうというアドレスを使っていますが、いまのところ、壊れることもないので大丈夫だと思います。

BASICインタープリタがあるので6000番地からにしてありますが、アセンブラを持っている方はアセンブル・リストを打ち込んでORG命令のアドレスを変えれば他のアドレスにも移せます。リレータを使ってもアドレス変換できると思いますが、やっていないので何とも言えません。

80Bのテープの音を聞いたことがありますか？ プログラムの始まる前に「カバッ」というような面白い音がします。ここがインフォメーション・ブロックの所です。その前後に長々と空白が必要なので、80Bは2,000ボーといいいながら、けっこうSAVEやLOADの時間は長くなっています。その代わりプログラムを探すまでは早送りをしてくれるので、大変便利にできています。

□参考文献

- 1) 高田嘉治・佐藤英昭：「PC→MZ変換プログラム」, I/O, '81年7月号
- 2) 大垣泰二：「MZ→PC変換プログラム」, I/O, '80年10月号



コンバータ・プログラム・リスト

```

0000 MON1: EDU 0000H
002E LETNL: EDU 0A2EH
0089 MSG: EDU 08B9H
004CE CMTSTP: EDU 04CEH
0072 KEYIN: EDU 0B32H
0057 MOTOR: EDU 0457H

:
: ORG 6000H

6000 11C660 LD DE, DATA1
6007 CDB908 CALL MSG
6006 CD2E0A CALL LETNL
6009 CD2E0A CALL LETNL
600C CD5704 CALL MOTOR
600F 11DE60 LD DE, DATA2
6012 CDB908 CALL MSG
6015 CD2E0A CALL LETNL
6018 CD5760 CALL MAIN
601B 2812 JR C, MG4
601D CDCE04 CALL CMTSTP
6020 11EE60 LD DE, DATA4
6023 CDB908 MG2: CALL MSG
6026 CD2208 MG3: CALL KEYIN
6029 FE0D CP 0DH
602B 20F9 JR NZ, MG3
602D C30000 JP MON1

:
: MG4: CALL CMTSTP
6030 CDCE04 LD DE, DATA5
6033 11FC60 JR MG2
6036 18E8

:
: NOP

6038 00

:
: MAIN: CALL SELECT
603C 18FB JR C, MAIN
603E CDAB60 MA1: CALL LDBAT
6041 FE3A CP 3AH
6043 20F9 JR NZ, MA1
6045 CDAB60 CALL LDBAT
6048 67 LD H, A
6049 4F LD C, A
604A CDAB60 CALL LDBAT
604D 6F LD L, A
604E B1 ADD A, C
604F 4F LD C, A
6050 CDAB60 CALL LDBAT
6053 B1 ADD A, C
6054 2018 JR NZ, MA4
6056 CDAB60 MA2: CALL LDBAT
6059 FE3A CP 3AH
605B 2014 JR NZ, MA4
605D CDAB60 CALL LDBAT
6060 47 LD B, A
6061 4F LD C, A
6062 A7 AND A
6063 CB RET Z
6064 CDAB60 MA3: CALL LDBAT
6067 77 LD (HL), A
606B B1 ADD A, C
6069 4F LD C, A
606A 23 INC HL
606B 05 DEC B
606C 20F6 JR NZ, MA3
606E CDAB60 CALL LDBAT
6071 B1 MA4: ADD A, C
6072 28E2 JR Z, MA2
6074 37 SCF
6075 C9 RET

:
: SELECT: IN A, (0E1H)
6076 DBE1 AND 40H
607A E640 JR Z, SELECT
607C 1E00 LD E, 0
607E 1C INC E
607F DBE1 ST1: IN A, (0E1H)

```

```

6081 E640 AND 40H
6083 20F9 JR NZ, ST1
6085 7B LD A, E
6086 FE20 CP 20H
608B C9 RET

:
6089 CD7660 LDBIT: CALL SELECT
608C 3806 JR C, LI1
608E 0E00 LD C, 0
6090 0601 LD B, 1
6092 1804 JR LI2

:
6094 0E01 LI1: LD C, 1
6096 0603 LD B, 3
609B DBE1 LI2: IN A, (0E1H)
609A E640 AND 40H
609C 28FA JR Z, LI2
609E DBE1 LI3: IN A, (0E1H)
60A0 E640 AND 40H
60A2 20FA JR NZ, LI3
60A4 10F2 DJNZ LI2
60A6 79 LD A, C
60A7 C9 RET

:
60AB C5 LDBAT: PUSH BC
60A9 D5 PUSH DE
60AA E5 PUSH HL
60AB CDB960 LB1: CALL LDBIT
60AE A7 AND A
60AF 20FA JR NZ, LB1
60B1 210008 LD HL, 0B00H
60B4 CDB960 LB2: CALL LDBIT
60B7 A7 AND A
60B8 2801 JR Z, LB3
60BA 37 SCF
60BB 7D LB3: LD A, L
60BC 1F RRA
60BD 6F LD L, A
60BE 25 DEC H
60BF 20F3 JR NZ, LB2
60C1 7D LD A, L
60C2 E1 POP HL
60C3 D1 POP DE
60C4 C1 POP BC
60C5 C9 RET

:
60C6 2A2A2A20 DATA1: DB '*** PC-MZ CONVERTER
***', 0DH
60CA 50432D40
60CE 5A20434F

-----

60DE 4E564552
60DE 54455220
60DA 2A2A2A0D

:
60DE 4C4F4144 DATA3: DB 'LOADING PC-TAPE', 0DH
60E2 494E4720
60E6 50432D54
60EA 4150450D

:
60EE 4F4B203A DATA4: DB 'OK : CR = MON', 0DH
60F2 20435220
60F6 3D204D4F
60FA 4E0D

:
60FC 42414420 DATA5: DB 'BAD : CR = MON', 0DH
6100 3A204352
6104 203D204D
610B 4F4E0D

:
610B END

```

MZB → PC CONVERSION



私の所属する大学の研究所に、ロボット制御用のCRC-80のプログラム開発機として、MZ-80Bが入りました。ところがMZBにはまだアセンブラがありませんでした。そこで私のPCの力を貸して、BASEをMZBに移植しました。そのとき大変役立ったのが9月号の“PC→MZBコンバータ”です。そこで今度は、MZBのBASEでPCのプログラムを作ろうと思い、その前に作ったのがこのプログラムです。

1

使用方法

初めに断っておきますが、これはMZBでPCが読み込めるテープを作るプログラムです。従って、このプログラムはMZB上で動くものです。また、いまのところオブジェクト・レベル（マシン語）のプログラムしか変換できません。さて、このプログラムはBASEで書いたものでBASEを持っている人は、ソースを入力して自分の通合のよいエリアにBASEし直すといでしょう。そうでない人は、左側のオブジェクトをモニタのMコマンドを使って入力してください。

\$5000からスタートすると、①開始番地、②終了番地、③オフセットを聞いてきますので、必ず16進4桁で入力してください。そうしない、また①から聞き直してきます。ここで④、⑤は、テープにSAVEする（PCに送る）プログラムの開始、終了番地のことです。⑥はたとえば、MZの\$6000からのプログラムをオフセットに\$3000を指定してSAVEしたとすれば、PCでLOADするときには、\$9000番地からメモリに入ります。この機能によりPCではROM領域のアドレスのプログラムも、ブロック転送することなくPCに送ることができます。オフセットが0のときでも入力は4桁で（0000）行なってください。

入力が終了し[Y]キーを押せば書き込みが始まります。このとき、テープに書き込んでいるのと同じ音をスピーカでモニタできます。

書き込みが終了したら後はPCのモニタのLコマンドで読み込みます。

2

プログラムについて

PCのテープ・フォーマットについては、マーク2,400Hz、スペース1,200HzのFSK方式600ボーとしかわからなかったもので、81年7月号のPC→MZ変換プログラムを解析して逆にPCのフォーマットを予測しました。初め、各バイトをスタート・ビット（スペース）1ビット+データ（下位より）8ビット+ストップ・ビット（マーク）1ビットの合計10ビットだと思い、そのようなプログラムを作りました。ところがそれで作ったテープは、数10バイトなら読むのですが、数Kバイトのテープはエラーばかりで読みません。

そこで、もう一度考え直したところ、ストップ・ビットが2ビットなのではないかという結論に達しました。そうしてみると、今度はテープデッキを使用し、トーンやボリュームをシビアに調整すれば読むようになりました。

しかし私がPC用に使用しているテレコでは読みません。音を聞いてみると高音がつぶれたいわゆるこもった音でした。これはMZのデッキと私のテレコのヘッド・アジマスとのずれによるものに違いありません。よくPCではほかのテレコでSAVEしたテープはLOADしにくいとか、オーディオの世界でもテープデッキ間の相性が問題になります。この原因はほとんどヘッドのアジマス変差が原因です。このアジマスとは、テープの走行方向に対するヘッドの角度というようにことで、録音時と再生時とでずれがあると、周波数特性が著しく劣化します。要するにこのために再生波形が乱れ、信号の立ち上がりがハッキリしなくなるとエラーを招く原因になっていたと考えられます。

そこでこれを少しでも防ぐために、信号のデューティ比を1:1でなくし、1に立ち上っている時間を少々長めにしてみました。この結果、MZのデッキと非常に相性の悪い私のテレコでもまったくエラーがでないテープを作れるようになったのです。みなさんのテレコでもまずだいじょうぶでしょう。

私はサイログ表記のアセンブラに慣れ親しんでいたのですが、BASEなど邪道だと思っていました。ところが今回ソース・リストが発表されている利点からMZBに移話し、使ってみてそのすばらしさに驚きました。もう他のアセンブラは使えなかりしいくらいです。なんとDO→UNTIL女のネスティングまでできます。ただ2つ気になる点を言うと、DO

→UNTIL女で制御変数にBレジスタ以外のものを使うと必ず絶対ジャンプになってしまうこと。また実際にメモリに落とさずBASEリストのみ出力することができないの2点です。まあ、たいした問題ではなく、利点がそれら欠点をカバーして余りあるものだと思います。使っていない方は早く使ってみることをお勧めします。

それでは今回はこのへんで失礼します。

参考文献

- 1) 高田房治・佐藤英昭:「P C→M Z変換プログラム」, I/O '81年7月号

MZB→PC CONVERTER プログラム・リスト

```

51DA 1 ;*****
51DA 2 ; MZB + PC TAPE CONVERTER
51DA 3 ;
51DA 4 ;
51DA 5 ;*****
51DA 6 ;
5000 7 START #5000
5000 8 ;
5000 9 MDN EQU #B1
5000 10 LETN EQU #AZE
5000 11 MSG EQU #B99
5000 12 BELL EQU #F14
5000 13 GETL EQU #6A4
5000 14 BUF EQU #1093
5000 15 HLX EQU #614
5000 16 PEYN EQU #B32
5000 17 MOTOR EQU #457
5000 18 MSTOP EQU #4CE
5000 19 ;
5000 20 CALL LETN:DE=TITLE
5000 21 CALL MSG
5000 22 BEGIN CALL BELL:CALL LETN
5000 23 CALL LETN:DE=TOPADD
5000 24 CALL MSG
5000 25 CALL LETN:DE=BUF
5000 26 CALL GETL
5000 27 CALL HLX:HL=(STADD)+HL
5000 28 IF CY GOTO BEGIN
5000 29 (DE=DE+ENDADD)
5000 30 CALL MSG
5000 31 JDE:CALL LETN
5000 32 CALL GETL
5000 33 CALL HLX:IF CY GOTO BEGIN
5000 34 DE=(STADD)
5000 35 HL=HL-DE+HL+
5000 36 (STBYT)+HL
5000 37 CALL LETN:DE=OFFSET
5000 38 CALL MSG
5000 39 CALL LETN:DE=BUF
5000 40 CALL GETL
5000 41 CALL HLX:IF CY GOTO BEGIN
5000 42 DE=(STADD)+HL=HL+DE
5000 43 (SVADD)+HL
5000 44 DE=READY:CALL MSG
5000 45 WAIT CALL KEYIN
5000 46 IF A=N GOTO MON
5000 47 IF A<0 Y GOTO WAIT
5000 48 CALL LETN:CALL MOTOR
5000 49 DE=NRMSG:CALL MSG
5000 50 DINT
5000 51 LEAD DC BC=#2000
5000 52 CALL MARK
5000 53 UNTIL DEC(B)=0
5000 54 ADRS A=1:CALL BYTE
5000 55 A=(SVADD-1):C=A
5000 56 CALL BYTE
5000 57 A=(SVADD):CALL BYTE
5000 58 A=A+C:NEG:CALL BYTE
5000 59 HL=(STADD)
5000 60 DATA DD DE=(STBYT)
5000 61 A=1:CALL BYTE
5000 62 A=D:IF A<0 GOTO NOMAL
5000 63 A=E:IF A=FF GOTO NOMAL
5000 64 CALL BYTE:C=A:GOTO LAST
5000 65 NOMAL A=FF:C=A:CALL BYTE
5000 66 LAST DD B=H
5000 67 A=(HL+1):CALL BYTE:DE=
5001 81 AF
5003 10 F6
5005 ED 44 CD 0A 51
500A 7A B3 C2 80 50
500F 3E 3A CD 0A 51 AF
5016 CD 0A 51 CD 0A 51
501B 01 00 05
5020 CD 44 51
5027 0B 7B B1 C2 EE 50
502F CD CE 0A FB
5036 CD 14 0F
503F CD 2E 0A 11 CF 51
5044 CD 89 0B C3 B1 00
5049 D9 AF 0B
5050 CD 25 51
5056 0A 00
505B CD 09 DC 44 51
5060 D7 24 51
5066 1A 10 F6
506B CD 44 51 CD 44 51
5070 CD D9 C9
5077 1E 02
507C 3E 93 E2
5081 3E 66 32 64 51
5088 CD 63 51
508D 3E 18 03 E2
5092 3E 61 32 64 51
5099 CD 63 51
50A6 D2 27 51
50AB C9
50B0 1E 04
50B6 3E 9C 03 E2
50BB 3E 32 64 51
50C0 CD 63 51
50C6 3E 18 03 E2
50CB 3E 61 32 64 51
50D0 CD 63 51
50D6 D2 27 51
50DB C9
50E0 1E 04
50E6 3E 9C 03 E2
50EB 3E 32 64 51
50F0 CD 63 51
50F6 D2 27 51
5103 C9
5109 1E 04
510E 3E 61 32 64 51
5113 CD 63 51
5119 D2 27 51
511E C9
5123 3E 2A 2A 2A 20 4D
5129 5A 2F 42 20 9B 83
5136 30 50 43 20
5143 20 54 01 50 45 20
5150 43 4F 4E 56 45 32
5156 5A 45 52 20 2A 2A
5163 2A 00
5169 53 54 51 52 54 20
5176 41 44 44 30 3F 0D
5183 4E 4E 44 20 20 20
5190 41 44 44 30 3F 0D
5196 46 46 53 45 54
51A3 20 3F 0D
51A9 52 65 61 64 79 20
51B6 3F 20 29 3F 2F 4E
51C3 29 20 0D
51C9 52 57 72 69 74 69
51D6 67 20 50 43 20
51E3 61 70 45 20 0D
51F0 20 4F 48 20 0D
51FD 00 00
51FA 00 00
51FF 00 00
5204 00 00
520B 00 00
5210 00 00
5217 00 00
5224 00 00
5231 00 00
5238 00 00
5245 00 00
5252 00 00
5259 00 00
5266 00 00
5273 00 00
5280 00 00
5287 00 00
5294 00 00
5301 00 00
5308 00 00
5315 00 00
5322 00 00
5329 00 00
5336 00 00
5343 00 00
5350 00 00
5357 00 00
5364 00 00
5371 00 00
5378 00 00
5385 00 00
5392 00 00
5399 00 00
5406 00 00
5413 00 00
5420 00 00
5427 00 00
5434 00 00
5441 00 00
5448 00 00
5455 00 00
5462 00 00
5469 00 00
5476 00 00
5483 00 00
5490 00 00
5497 00 00
5504 00 00
5511 00 00
5518 00 00
5525 00 00
5532 00 00
5539 00 00
5546 00 00
5553 00 00
5560 00 00
5567 00 00
5574 00 00
5581 00 00
5588 00 00
5595 00 00
5602 00 00
5609 00 00
5616 00 00
5623 00 00
5630 00 00
5637 00 00
5644 00 00
5651 00 00
5658 00 00
5665 00 00
5672 00 00
5679 00 00
5686 00 00
5693 00 00
5700 00 00
5707 00 00
5714 00 00
5721 00 00
5728 00 00
5735 00 00
5742 00 00
5749 00 00
5756 00 00
5763 00 00
5770 00 00
5777 00 00
5784 00 00
5791 00 00
5798 00 00
5805 00 00
5812 00 00
5819 00 00
5826 00 00
5833 00 00
5840 00 00
5847 00 00
5854 00 00
5861 00 00
5868 00 00
5875 00 00
5882 00 00
5889 00 00
5896 00 00
5903 00 00
5910 00 00
5917 00 00
5924 00 00
5931 00 00
5938 00 00
5945 00 00
5952 00 00
5959 00 00
5966 00 00
5973 00 00
5980 00 00
5987 00 00
5994 00 00
6001 00 00
6008 00 00
6015 00 00
6022 00 00
6029 00 00
6036 00 00
6043 00 00
6050 00 00
6057 00 00
6064 00 00
6071 00 00
6078 00 00
6085 00 00
6092 00 00
6099 00 00
6106 00 00
6113 00 00
6120 00 00
6127 00 00
6134 00 00
6141 00 00
6148 00 00
6155 00 00
6162 00 00
6169 00 00
6176 00 00
6183 00 00
6190 00 00
6197 00 00
6204 00 00
6211 00 00
6218 00 00
6225 00 00
6232 00 00
6239 00 00
6246 00 00
6253 00 00
6260 00 00
6267 00 00
6274 00 00
6281 00 00
6288 00 00
6295 00 00
6302 00 00
6309 00 00
6316 00 00
6323 00 00
6330 00 00
6337 00 00
6344 00 00
6351 00 00
6358 00 00
6365 00 00
6372 00 00
6379 00 00
6386 00 00
6393 00 00
6400 00 00
6407 00 00
6414 00 00
6421 00 00
6428 00 00
6435 00 00
6442 00 00
6449 00 00
6456 00 00
6463 00 00
6470 00 00
6477 00 00
6484 00 00
6491 00 00
6498 00 00
6505 00 00
6512 00 00
6519 00 00
6526 00 00
6533 00 00
6540 00 00
6547 00 00
6554 00 00
6561 00 00
6568 00 00
6575 00 00
6582 00 00
6589 00 00
6596 00 00
6603 00 00
6610 00 00
6617 00 00
6624 00 00
6631 00 00
6638 00 00
6645 00 00
6652 00 00
6659 00 00
6666 00 00
6673 00 00
6680 00 00
6687 00 00
6694 00 00
6701 00 00
6708 00 00
6715 00 00
6722 00 00
6729 00 00
6736 00 00
6743 00 00
6750 00 00
6757 00 00
6764 00 00
6771 00 00
6778 00 00
6785 00 00
6792 00 00
6799 00 00
6806 00 00
6813 00 00
6820 00 00
6827 00 00
6834 00 00
6841 00 00
6848 00 00
6855 00 00
6862 00 00
6869 00 00
6876 00 00
6883 00 00
6890 00 00
6897 00 00
6904 00 00
6911 00 00
6918 00 00
6925 00 00
6932 00 00
6939 00 00
6946 00 00
6953 00 00
6960 00 00
6967 00 00
6974 00 00
6981 00 00
6988 00 00
6995 00 00
7002 00 00
7009 00 00
7016 00 00
7023 00 00
7030 00 00
7037 00 00
7044 00 00
7051 00 00
7058 00 00
7065 00 00
7072 00 00
7079 00 00
7086 00 00
7093 00 00
7100 00 00
7107 00 00
7114 00 00
7121 00 00
7128 00 00
7135 00 00
7142 00 00
7149 00 00
7156 00 00
7163 00 00
7170 00 00
7177 00 00
7184 00 00
7191 00 00
7198 00 00
7205 00 00
7212 00 00
7219 00 00
7226 00 00
7233 00 00
7240 00 00
7247 00 00
7254 00 00
7261 00 00
7268 00 00
7275 00 00
7282 00 00
7289 00 00
7296 00 00
7303 00 00
7310 00 00
7317 00 00
7324 00 00
7331 00 00
7338 00 00
7345 00 00
7352 00 00
7359 00 00
7366 00 00
7373 00 00
7380 00 00
7387 00 00
7394 00 00
7401 00 00
7408 00 00
7415 00 00
7422 00 00
7429 00 00
7436 00 00
7443 00 00
7450 00 00
7457 00 00
7464 00 00
7471 00 00
7478 00 00
7485 00 00
7492 00 00
7499 00 00
7506 00 00
7513 00 00
7520 00 00
7527 00 00
7534 00 00
7541 00 00
7548 00 00
7555 00 00
7562 00 00
7569 00 00
7576 00 00
7583 00 00
7590 00 00
7597 00 00
7604 00 00
7611 00 00
7618 00 00
7625 00 00
7632 00 00
7639 00 00
7646 00 00
7653 00 00
7660 00 00
7667 00 00
7674 00 00
7681 00 00
7688 00 00
7695 00 00
7702 00 00
7709 00 00
7716 00 00
7723 00 00
7730 00 00
7737 00 00
7744 00 00
7751 00 00
7758 00 00
7765 00 00
7772 00 00
7779 00 00
7786 00 00
7793 00 00
7800 00 00
7807 00 00
7814 00 00
7821 00 00
7828 00 00
7835 00 00
7842 00 00
7849 00 00
7856 00 00
7863 00 00
7870 00 00
7877 00 00
7884 00 00
7891 00 00
7898 00 00
7905 00 00
7912 00 00
7919 00 00
7926 00 00
7933 00 00
7940 00 00
7947 00 00
7954 00 00
7961 00 00
7968 00 00
7975 00 00
7982 00 00
7989 00 00
7996 00 00
8003 00 00
8010 00 00
8017 00 00
8024 00 00
8031 00 00
8038 00 00
8045 00 00
8052 00 00
8059 00 00
8066 00 00
8073 00 00
8080 00 00
8087 00 00
8094 00 00
8101 00 00
8108 00 00
8115 00 00
8122 00 00
8129 00 00
8136 00 00
8143 00 00
8150 00 00
8157 00 00
8164 00 00
8171 00 00
8178 00 00
8185 00 00
8192 00 00
8199 00 00
8206 00 00
8213 00 00
8220 00 00
8227 00 00
8234 00 00
8241 00 00
8248 00 00
8255 00 00
8262 00 00
8269 00 00
8276 00 00
8283 00 00
8290 00 00
8297 00 00
8304 00 00
8311 00 00
8318 00 00
8325 00 00
8332 00 00
8339 00 00
8346 00 00
8353 00 00
8360 00 00
8367 00 00
8374 00 00
8381 00 00
8388 00 00
8395 00 00
8402 00 00
8409 00 00
8416 00 00
8423 00 00
8430 00 00
8437 00 00
8444 00 00
8451 00 00
8458 00 00
8465 00 00
8472 00 00
8479 00 00
8486 00 00
8493 00 00
8500 00 00
8507 00 00
8514 00 00
8521 00 00
8528 00 00
8535 00 00
8542 00 00
8549 00 00
8556 00 00
8563 00 00
8570 00 00
8577 00 00
8584 00 00
8591 00 00
8598 00 00
8605 00 00
8612 00 00
8619 00 00
8626 00 00
8633 00 00
8640 00 00
8647 00 00
8654 00 00
8661 00 00
8668 00 00
8675 00 00
8682 00 00
8689 00 00
8696 00 00
8703 00 00
8710 00 00
8717 00 00
8724 00 00
8731 00 00
8738 00 00
8745 00 00
8752 00 00
8759 00 00
8766 00 00
8773 00 00
8780 00 00
8787 00 00
8794 00 00
8801 00 00
8808 00 00
8815 00 00
8822 00 00
8829 00 00
8836 00 00
8843 00 00
8850 00 00
8857 00 00
8864 00 00
8871 00 00
8878 00 00
8885 00 00
8892 00 00
8899 00 00
8906 00 00
8913 00 00
8920 00 00
8927 00 00
8934 00 00
8941 00 00
8948 00 00
8955 00 00
8962 00 00
8969 00 00
8976 00 00
8983 00 00
8990 00 00
8997 00 00
9004 00 00
9011 00 00
9018 00 00
9025 00 00
9032 00 00
9039 00 00
9046 00 00
9053 00 00
9060 00 00
9067 00 00
9074 00 00
9081 00 00
9088 00 00
9095 00 00
9102 00 00
9109 00 00
9116 00 00
9123 00 00
9130 00 00
9137 00 00
9144 00 00
9151 00 00
9158 00 00
9165 00 00
9172 00 00
9179 00 00
9186 00 00
9193 00 00
9200 00 00
9207 00 00
9214 00 00
9221 00 00
9228 00 00
9235 00 00
9242 00 00
9249 00 00
9256 00 00
9263 00 00
9270 00 00
9277 00 00
9284 00 00
9291 00 00
9298 00 00
9305 00 00
9312 00 00
9319 00 00
9326 00 00
9333 00 00
9340 00 00
9347 00 00
9354 00 00
9361 00 00
9368 00 00
9375 00 00
9382 00 00
9389 00 00
9396 00 00
9403 00 00
9410 00 00
9417 00 00
9424 00 00
9431 00 00
9438 00 00
9445 00 00
9452 00 00
9459 00 00
9466 00 00
9473 00 00
9480 00 00
9487 00 00
9494 00 00
9501 00 00
9508 00 00
9515 00 00
9522 00 00
9529 00 00
9536 00 00
9543 00 00
9550 00 00
9557 00 00
9564 00 00
9571 00 00
9578 00 00
9585 00 00
9592 00 00
9599 00 00
9606 00 00
9613 00 00
9620 00 00
9627 00 00
9634 00 00
9641 00 00
9648 00 00
9655 00 00
9662 00 00
9669 00 00
9676 00 00
9683 00 00
9690 00 00
9697 00 00
9704 00 00
9711 00 00
9718 00 00
9725 00 00
9732 00 00
9739 00 00
9746 00 00
9753 00 00
9760 00 00
9767 00 00
9774 00 00
9781 00 00
9788 00 00
9795 00 00
9802 00 00
9809 00 00
9816 00 00
9823 00 00
9830 00 00
9837 00 00
9844 00 00
9851 00 00
9858 00 00
9865 00 00
9872 00 00
9879 00 00
9886 00 00
9893 00 00
9900 00 00
9907 00 00
9914 00 00
9921 00 00
9928 00 00
9935 00 00
9942 00 00
9949 00 00
9956 00 00
9963 00 00
9970 00 00
9977 00 00
9984 00 00
9991 00 00
10000 00 00

```

MZ/KC → MZB マシン語 CONVERTER



■野口岳郎とNeptune Soft

（株）I/Oエンジンルーム

NEPTUNE SOFTが出す、MZB用ソフト第3弾は、何と旧MZのマシン語をMZBで走るように変換してしまう、MZ K/C → MZBマシン語コンバータです。

1 目的

BASICのプログラムのコンバータはMZBの発売と同時に販売されているのに、マシン語の方は未だに逆アセンブルでししながら番地対照表をにらみ、1つ1つ手で書き変えなければなりません。これには膨大な労力と時間がかかる上、人間のやることですからミスもよく起こります。そこで、マシン語をB用に変換するプログラムを作ったのです。

2 使用感

\$F700番地からスタートさせると、タイトルを表示してコマンド待ちになります。

[L] キーを押すと、

Tape Set Ok?

と表示されるので、K/C用のマシン語のテープをセットし、[CR] を押してください。各種データを表示し、テープを読み終ると再びコマンド待ちになります。

コマンド表

- L … K/C用のマシン語をロードします。
- D … データ・エリアの設定に使います。
- C … コンバート・コマンドです。
- ! … モニタに戻ります。

マシン語中にデータ・エリアがある場合は、[D] コマンドで登録してください。

上のようになりますから、左から順に登録番号（0からFまで、必ず0から始めること）、データ・エリアのトップ、エンドの順に書き込み、[CR] で登録します。登録が済んだら、[CR] だけを押すとコマンド待ちになりますから、もう一度 [D] で、新しく登録されたか確かめてください。データ・エリアがわからないときは [!] で一応モニタに戻ってデータ・エリアを確認してください。ホット・スタートは \$F71F、コールド・スタートは \$F700です。なお、[C]、[D] コマンドはテープを読んだ後でないと受け付けません。

さて、次はいよいよコンバートです。[C] キーを押すと自動的にコンバートを始めます。特に長いものでなければ、ほとんど一瞬で完了します。コンバートが終わると、また、

Tape Set Ok?

と出るので、新しいテープをセットして [CR] を押すと、変換されたプログラムが2000ボーでSAVEされます。

3 プログラム

何のことはありません。モニタ・コールの番地を書き変えていただけです。ただし、\$09B3などBにないルーチンをCALLしているものは、基走防止のためRST 38Hをかけています。ですから、もし途中で急に、×××× ××××……と16進数を表示してモニタに戻ったときはこれに当たったときです。一番右のプログラム・カウンタの値を参考に、適切な変更をしてください。

なお、\$0047、\$0DA6については、Bには必要なしと思われるためNOPで埋めています。

また、このプログラムではワーク・エリア（\$1000～\$11FF）をアクセスしていると思われるものについても変換しています。おかしな所をいじっているものは、これまたRST 38Hをかけます。ですから、\$22、2A、32、3Aはもちろん、\$21（HLレジスタによるメモリ・アクセスが多い）、\$31（スタック・ポインタをワーク・エリアに置くケースが多い）、\$ED、\$43とか\$FD、\$21といったものまで変換しています。\$11については独自の変換をしています。そのため、ごく一般的なプログラムであれば変換しただけで正常に動作します。

4 追 伸

これで、MZ用の膨大なソフトウェアが簡単にB用に流用できるようになれば幸いです。判断基準の誤りなどがありましたら、ご連絡くださいませ。なにぶん、「まじめ」な1Kバイト以上のマシン語を組むのは初めてなので…。

最後になりましたが、MZ-80K2/Bを快くデモらせてくださいました、久留米カホパーツとベスト電器の皆様、この場を借りてお礼申し上げます。

変換時の注意 エンジン

MZ K/C用のプログラムで\$1200から始まっているもの



MZ-80B



VOICE MZBで音声入出力を! SYSTEM/80B

●三屋一彦

MZ 80Bを購入したのが6月。BASICのプログラミングにいや気がして、もっぱらマシン語でプログラミング(特にシステム関係)をしています。

MZ 80K Cの音声入出力システムの記事を見て、K/CにできてBにできないはずがないと思い立ち、B用の「VOICE SYSTEM」を発表します。

ハードウェア

MZ 80Bのハードの改造は不要。

ソフトウェア

ダンブ・リストのとおりで、E000H～F7BFHまでの約5.8KバイトでE000H～E3BFHがプログラム、残りは音声データです。リロケートは「S.R.D」のRコマンドで簡単にできます。

使い方

モニタから、Jコマンドで次のアドレスからスタートさせていただきます。

① キー デモンストレーション(E006H)

キー入力待ちになるので、0～9、A～Fのキーを押すと、その音声が入蔵スピーカから聞こえます。

② VOICE DUMP(E003H)

アドレスを聞いてくるので、ダンブしたい先頭アドレスを入力してください。アドレスとメモリの内容を画面に表示しながら、読み上げていきます。**[BREAK]**キーでモニタに戻ります。

③ VOICE WRITE(E000H)

アドレスを聞いてくるので、書き込みたい先頭アドレスを入力してください。

アドレスを表示して読み上げた後、キー入力待ちになります。16進数を入力すると読み上げながらメモリに書き込んでいきます。

誤入力は、カーソルの左移動キーで修正できます。**[CR]**キーまたは**[BREAK]**キーでモニタに戻ります。

④ 音声テスト入力(E060H)

音声または音楽などを普通のラジカセなどで録音し、

MZ 80B本体のカセットにセットして、E060Hにジャンプさせてください。

約30秒間、音声入力を行なって、モニタに戻ります。音声データは12A0HからDFFFHまでの間に書き込まれています。

⑤ 音声テスト出力(E065H)

上記で入力された音声データを出力します。

プログラムの変更

① **[BREAK]**キーでモニタに戻りますが、E00EHの2バイトを書き換えれば任意のアドレスにジャンプできます。

② VOICE DUMPの無音の待ち時間はE217Hからの2バイトを変えれば調節できます(現在は01F0H)。

③ 音声テスト入出力における音声データの収納範囲(12A0H～DFFFH)はE06CH、E1D8H(先頭アドレス、現在は12A0H)およびE06FH、E1DBH(バイト数、現在はCD60H)の内容を変えれば変更できます。

④ 音声出力のボーレートはE04EH(現在は0FH)の値、音声入力のボーレートはE01FH(現在は11H)の値を変えれば変更できます。小さくすると速くなります。

⑤ E3C0H～F6BFHの音声データは0～Fの順に140Hごとに収納されています。

あとがき

MZ-80K/Cの音声は聞いたことがないので何とも言えません、音質はまあまあです。

このプログラムをマシン語の入力、デバッグに役立ててください。

最後になりましたが、音声データの作成時に、声のデータをいただいた名古屋工業大学生活協同組合の織田道子さんに深く感謝いたします。

P.S.

VOICE DUMP時に**[SPACE]**キーを押すと一時停止します。モニタはSB 1520です。

□参考文献

ツールバグ・コンピュータ・サービス:「MZ-80 音声出力システム」, I/O, '81年8月号

VOICE SYSTEM ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
EA00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EA10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EA20	00	00	00	00	00	00	00	00	00	00	00	00	00	38	3C	38	180
EA30	1C	1E	0F	83	C1	C1	F0	7C	3B	64	00	1E	03	C0	1C	0F	180
EA40	F0	F0	01	90	B5	B5	0E	04	27	21	21	21	21	78	1F	1F2	180
EA50	0F	02	07	B1	E1	F0	7C	1B	1C	3F	27	07	08	03	80	162	180
EA60	01	98	01	1B	03	99	09	01	D8	09	D8	08	08	D8	B3	180	180
EA70	03	F2	13	DA	03	F0	17	F0	07	C0	0F	C0	1F	3F	80	180	180
EA80	3F	00	7F	00	C1	F0	07	F0	17	F0	07	00	5E	C0	1F	185	180
EA90	01	FF	00	7C	07	C1	F0	07	00	3C	30	0F	F8	B0	3E	03	106
EA0A	E2	0F	8F	0F	B3	03	7E	00	1F	01	F0	07	7C	07	C0	125	106
EA1A	01	F0	1F	00	07	C0	7C	00	1F	01	F0	00	FC	07	C0	03	129
EA2A	F0	1F	00	0F	3F	00	1F	00	7E	00	7E	00	3E	7C	00	7C	190
EA3A	00	7C	00	7C	07	00	00	06	04	1E	0F	83	C1	E0	F8	3C	1A0
EA4A	B1	E0	70	00	70	00	00	00	00	00	00	00	00	00	00	00	162
EA5A	0C	1E	07	83	E1	F0	7C	07	07	07	07	00	40	30	00	00	162
Sum	BF	22	B8	22	70	0C	67	C3	D6	AD	C4	F3	F3	22	2E	C0	19E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
EB00	3B	01	00	00	00	00	04	00	00	00	00	00	00	00	00	00	130
EB10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB50	00	00	00	00	04	00	00	00	00	00	00	00	00	00	20	03	127
EB60	00	00	00	01	C2	00	01	B6	00	00	1B	00	00	00	00	00	16E
EB70	03	10	03	10	03	40	00	00	00	00	00	01	8C	3B	00	38	177
EB80	40	00	3B	C0	01	C3	08	00	00	00	00	00	00	1C	01	80	1C1
EB90	00	00	00	00	00	00	C0	02	00	00	80	00	00	00	00	00	142
EB0A	30	60	00	00	70	00	00	00	00	00	00	00	00	00	00	00	100
EB1A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB0B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB1B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB0C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB1C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB0D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB1D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB0E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB1E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB0F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB1F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB2F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB3F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB4F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB5F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB6F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB7F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB8F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EB9F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
Sum	CB	71	38	F1	73	C7	CE	0D	80	03	7F	41	26	43	21	F9	15D

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
EC00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EC10	00	7B	60	F0	0F	08	7E	01	E0	1E	18	7C	01	C0	3C	70	F8
EC20	03	CA	0F	00	F0	03	08	3E	03	08	5F	01	F0	3E	03	4B	1C4
EC30	17	81	0F	07	C0	F0	0E	03	1E	07	C0	0E	39	07	7C	1E	1C8
EC40	07	1C	0F	81	C1	F1	C0	00	7A	3C	1E	07	07	07	07	07	162
EC50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EC60	00	7C	1C	0F	00	0F	81	80	C0	01	F0	70	1C	1E	0E	3B	1C8
EC70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
EC80	3F	C0	00	00	7F	F0	00	00	27	E0	00	00	99	8B	00	02	C0
EC90	07	EB	09	90	13	E0	C1	C0	19	80	19	0E	19	C0	3C	0C	1A6
EC90	19	81	38	C0	19	83	88	C0	87	83	30	05	83	70	01	01	100
ECB0	00	00	00	00	00	00	00	00	7A	00	00	00	00	00	00	00	100
ECB0	03	08	0E	F4	00	08	00	00	00	00	00	00	00	00	00	00	100
ECB0	BC	3B	03	0C	61	0F	81	84	B6	7E	E7	1C	3F	87	38	61	1D1
ECDB	FC	39	02	07	C1	9C	00	7C	00	00	00	00	00	00	00	00	100
ECF0	F0	07	C0	1F	00	7C	C1	F8	07	80	1F	80	F0	03	38	0F	1B3

VOICE SYSTEM ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
F400	00	00	00	00	00	00	00	00	00	00	00	00	07	00	00	00	107
F410	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
F420	00	00	00	00	00	00	02	08	00	00	00	00	00	00	10	00	126
F430	00	04	0C	00	20	00	00	60	00	00	00	00	00	00	40	33	103
F440	30	00	00	1B	33	60	01	82	66	01	3B	32	66	42	12	1B0	
F450	37	6C	CC	66	66	4D	B1	09	00	9A	4C	80	B5	BD	27	60	141
F460	42	53	B3	20	31	25	93	00	00	66	93	20	03	67	B6	60	1EA
F470	32	6F	26	40	24	CE	4C	C0	9C	99	B1	9B	7B	32	03	103	
F480	36	F6	C4	04	C9	C9	13	37	26	64	6E	DB	80	93	11F		
F490	3B	62	6C	EC	BB	09	93	32	20	26	CE	DB	80	9B	33	1B7	
F4A0	60	06	CD	CB	B0	1B	37	26	00	4C	DC	90	01	33	72	44	19A
F4B0	04	C9	09	00	13	67	6C	00	49	89	B1	23	66	E4	C4	B0	1FD
F4C0	93	93	22	26	4E	BB	99	31	33	22	64	C4	CC	BD	9B	1C7	
F4D0	59	32	12	64	86	64	6C	C9	4C	C8	D9	92	C9	9B	99	26	10F
F4E0	D9	00	93	00	4B	8B	63	01	64	40	4C	93	13	26	26	49	1FB
F4F0	2C	99	89	93	69	22	62	66	C1	64	C0	0B	99	C0	BB	09	157
Sum	A1	B7	6D	47	0B	C9	5B	F7	53	EC	97	DA	B2	DA	C6	B2	143

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
F500	10	00	01	01	33	20	1B	33	12	26	05	B1	21	22	62	5B	131
F510	10	12	64	04	99	01	32	60	24	CB	0D	90	01	26	02	64	1CC
F520	CB	04	99	11	93	20	03	62	46	4C	C0	04	80	0B	91	01	1FE
F530	09	90	00	00	02	1B	00	00	06	06	59	00	00	00	00	49	15E
F540	00	00	00	00	00	00	01	09	00	06	12	99	80	00	36	DA	14B
F550	49	00	80	35	2D	20	92	42	24	AD	BB	DB	1B	18	6D	26	149
F560	B6	16	D2	40	24	BC	49	25	B2	4B	CB	84	92	C9	09	20	107
F570	B6	13	62	63	4B	44	89	35	25	60	4C	96	AD	B4	4B	96	14E
F580	A4	80	4B	07	60	00	C9	85	60	00	08	00	69	04	92	02	1B1
F590	4B	00	20	1E	90	00	01	35	26	02	60	60	22	6C	D6	1A5	
F5A0	CB	04	99	AD	80	09	31	BB	00	12	66	92	01	24	C7	24	1C1
F5B0	00	4D	BF	6C	00	99	1E	DB	05	92	05	80	0B	66	0D	20	1E1
F5C0	12	4C	D2	40	24	9B	36	00	49	30	2C	00	93	00	49	00	1E3
F5D0	20	42	92	00	00	85	80	01	0B	49	00	02	06	CB	04	152	
F5E0	04	94	90	00	00	69	20	00	00	12	40	00	00	24	80	00	1A7
F5F0	04	4B	00	00	0B	92	00	00	11	B4	00	20	22	4B	00	CB	1FD
Sum	B6	0A	C6	79	A3	06	D4	07	3A	62	3B	DD	05	DA	4C	B4	113

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
F600	C4	90	01	91	BB	20	12	02	36	C0	04	0C	6D	00	4B	9B	15B	
F610	92	00	80	31	24	00	60	42	4B	04	C0	B4	90	09	01	0D	170	
F620	20	11	00	0B	60	10	02	12	40	00	04	16	C0	00	04	12	1F0	
F630	40	00	00	52	4B	00	00	4B	4B	00	00	00	2C	00	00	16	1BC	
F640	94	00	00	0B	08	00	00	06	92	80	00	01	A4	80	00	40	1B4	
F650	6D	80	00	08	01	B4	00	00	01	12	00	00	00	03	68	00	12B	
F660	00	04	96	00	00	00	6D	B4	00	00	00	24	80	00	00	01	160	
F670	B6	90	00	00	01	20	DA	40	00	00	00	91	24	00	00	03	40	1B9
F680	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
F690	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	180	
F6A0	00	00	00	00	00	00	00	00	00	38	00	00	00	00	00	00	19B	
F6B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
F6C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
F6D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
F6E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
F6F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1C0	
Sum	6D	85	47	B3	EO	BE	21	5B	D1	D7	EC	DB	0D	AC	7B	CE	1A1	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
F700	00	00	00	00	00	80	00	20	01	00	00	00	00	00	00	1B	00	1B9
F710	00	07	20	01	00	26	00	23	01	0B	00	70	1B	30	9B	CC	199	
F720	F6	20	04	64	33	01	3F	80	DC	B4	F7	60	11	99	CC	CC	16C	
F730	67	39	9B	CC	46	70	CC	D6	62	33	91	CC	EC	57	62	13	146	
F740	99	23	D9	9B	CC	4C	B7	60	01	01	E6	61	33	19	CF	23	1B3	
F750	19	89	9C	D3	3B	CB	BD	C6	66	71	01	B6	CB	77	00	1F	150	
F760	B1	0E	CB	0E	FC	A1	F2	00	7E	00	7F	00	3F	00	1F	80	19F	
F770	1E	60	1F	80	CE	C0	77	00	39	80	FE	00	C0	00	7C	02	117	
F780	00	07	C1	30	00	1C	03	00	00	00	60	00	00	66	40	00	11D	
F790	04	C0	00	00	CC	C0	00	09	80	00	00	9B	00	00	19	00	1BA	
F7A0	00	01	B0	00	00	13	00	00	01	BB	00	00	00	20	00	01	19E	
F7B0	80	00	00	0B	C0	00	00	00	00	02	30	00	00	19	00	193		
Sum	62	42	8B	62	53	9B	AB	A9	DE	69	4E	BB	12	36	BA	70	195	

BASICのストリング変数を話す!

BASIC VOICE



●松井洋介

1/O '81年11月号の「BASIC VOICE」のMZ-80B版ができたので報告します。

ソフトウェア

メモリの関係上プログラムがB000H-B07DH、テーブルがB100H-B1FFHであり、これが今回変更した部分でB200H-ECBFHの音声データは11月号のものと同じです。

なお、カナ文字と数字小数点以外はすべて無音ということになります。アルファベットで何か音を出したいと思っている人は、ASCIIコード表とにらめっこしながら、テーブル内の01（無音）を書き変えてください。

使用方法

①. モニタでB000Hからプログラム、テーブル、データの順に入力し、テーブルにS-ADDR. B000H, E-ADDR. ECBF HとしてSAVEする。

②. BASICインタープリタ SB-5520を起動する。LIMIT\$ AFFFを実行してから、「BASIC VOICE」をLOADする。

③. USR (\$B000, A\$)をプログラムで実行するとA\$内のカタカナおよび数字をしゃべってくれます。また1文字終了ごとに[SIFT BREAK]キーが押されているかどうか調べています。

その他

小生はK/Cに関しては何も知らないのですが、11月号のプログラムを参考にして80B用に新しく作り直しました。しかし、全体としては11月号の物によく似ていると思います。

80Bでは8255のポートCにオーディオアンプが繋がっているの、このビットを立てたり、おろしたりして音を出しています。そこで、音声データをもとにビット・コントロール・モードで8255のビット操作をしています。

音程は\$B057の18Hを変えれば変化します。ピッチは\$B06Bの02Hを変えてください。ただし、00Hにはしないでください。30秒ほど待たされることになるので。

◆参考文献

1) ツールバグ・コンピュータ・サービス技術部:「ストリング変数をMZがしゃべる」, 1/O, '81年11月号

BASIC VOICE プログラム

B000 1B	DEC DE
B001 ED5E0B0	LD (B0E0H), DE
B005 2AE0B0	#B005:LD HL, (B0E0H)
B008 25	#B008:INC HL
B009 22E0B0	LD (B0E0H), HL
B00C 7E	LD A, (HL)
B00D FE0D	CP ODH
B00F CB	RET Z
B010 FEDE	CP DEH
B012 30F4	JR NC, #B00B
B014 0A00	LD B, 00H
B016 4F	LD C, A
B017 25	INC HL
B018 7E	LD A, (HL)
B019 FE0D	CP ODH
B01B 2B0E	JR Z, #B02B
B01D FEDE	CP DEH
B01F 2004	JR NZ, #B025
B021 0E30	LD B, 30H
B023 1B0A	JR #B02B
B025 FEDF	#B025:CP DFH
B027 2002	JR NZ, #B02B
B029 0A50	LD B, 50H
B02B 79	#B02B:LD A, C
B02C 80	ADD A, B
B02D 6F	LD L, A

B02E 2AB1	LD H, B1H
B030 7E	A, (HL)
B031 FE10	CP 10H
B033 21A000	LD HL, 00A0H
B036 3001	JR NC, #B039
B038 29	ADD HL, HL
B039 47	#B039:LD B, A
B03A EB	EX DE, HL
B03B 21C0B0	LD HL, #C00H
B03E 19	#B03E:ADD HL, DE
B03F 10FD	LD B, #B03E
B041 42	LD C, D
B042 4B	LD E, C
B043 C5	#B043:PUSH BC
B044 7E	LD A, (HL)
B045 0A08	LD B, 08H
B047 F5	#B047:PUSH AF
B048 F8B0	AND B0H
B04A 2B1C	JR Z, #B06B
B04C 3E05	LD A, 05H
B04E D7E3	OUT (E3H), A
B050 1B04	JR #B056
B052 7E04	#B052:LD A, 04H
B054 D7E3	OUT (E3H), A
B056 3E18	#B056:LD A, 18H
B058 3D	#B058:DEC A

B059 20FD	JR NZ, #B05B
B05B F1	POP AF
B05C 17	RLA
B05D 10E8	DJNZ RLA
B05F 25	JR #B047
B060 C1	POP BC
B061 0B	DEC BC
B062 7B	LD A, B
B063 B	OR C
B064 20DD	JR NZ, #B043
B066 1802	JR #B06A
B068 1BEB	#B068:JR #B052
B06A 0A02	B, 02H
B06C 11FFFH	LD DE, FFFFH
B06F 21DE39	#B06F:LD HL, 39DEH
B072 19	#B072:ADD HL, DE
B073 3BFD	JR C, #B072
B075 10F8	DJNZ #B06F
B077 C8A205	CALL 05A2H
B07A CB	RET Z
B07B C305B0	JR #B005
B07E 00	NOP
B080 00	NOP

テーブル

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F	Sum
B100 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B110 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B120 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	111
B130 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 01	151
B140 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B150 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B160 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B170 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B180 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110

B190 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B1A0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B1B0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B1C0 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37 38 39	108
B1D0 39 2A 3B 3C 3D 3E 3F 40 41 42 43 44 45 46 47 48 49	117
B1E0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	110
B1F0 51 52 53 54 55 56 57 58 59 5A 5B 5C 5D 5E 5F	110
Sum C2 DF EA E9 EE 05 06 0C 12 B8 C7 EE 80 E7 10C	

[illegible]

B500	00	01	+2	00	00	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	5Sun
B501	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B510	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B520	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B530	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B540	07	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B550	7F	80	07	00	00	98	12	46	08	00	5F	B3	0D	02	00	00	100
B560	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B570	FE	FE	FE	47	E0	30	02	00	FF	80	AF	80	1A	00	00	00	100
B580	FE	FE	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B590	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B600	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B610	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B620	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B630	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B640	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B650	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B660	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B670	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B680	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B690	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B700	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B710	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B720	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B730	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B740	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B750	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B760	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B770	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B780	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B790	00	00	00														

[illegible]

add	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Sum
B500	F8	00	F0	B0	3F	FC	07	B0	0E	F7	04	06	03	FC	10	25	158
B510	1C	00	05	FE	00	47	0C	90	00	00	01	FF	00	7F	B0	14E	159
B520	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0
B530	F8	01	FF	07	07	07	07	07	00	00	00	00	00	00	00	00	14F
B540	01	FF	00	3F	B0	7E	00	7F	0E	07	FB	00	FE	03	F0	3	1A5
B550	00	FF	00	07	FB	0F	0C	0F	FE	00	7F	3F	00	F0	3	F0	196
B560	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0
B570	03	F0	07	0F	00	3F	07	FC	07	E0	01	FC	07	F0	06	07	107
B580	0F	FC	07	01	FE	00	3E	00	00	00	07	0C	03	FB	06	00	108
B590	7F	00	1F	E0	1F	B0	03	00	00	FE	00	1F	FC	03	0C	00	12B
B5A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0
B5B0	00	00	00	00	00	00	0F	0C	0E	00	00	00	00	00	00	0E	1B8
B5C0	00	00	07	FF	30	00	00	00	7F	FF	0C	00	00	00	FF	FF	165
B5D0	00	00	00	00	00	00	00	00	00	00	F0	01	FF	FC	00	00	1E7
B5E0	3F	F0	01	FF	FF	00	00	1F	7F	00	00	00	00	00	00	00	1E8
B5F0	00	00	00	00	00	00	00	FE	00	00	EF	A0	00	01	FF	100	

[illegible]

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum
B700	1F	FF	C0	00	00	07	FF	FF	E0	00	00	07	FF	FF	FF	B0	147
B710	00	00	3F	FF	FF	FF	F0	00	00	00	0F	FF	FF	FE	00	00	137
B720	01	FF	FF	FF	FF	F0	00	03	FE	00	7F	FF	FF	C0	00	0A	135
B730	01	FF	F8	0F	FE	00	00	00	0F	FE	3F	F8	00	00	00	00	140

B740	FF	FB	00	7F	FC	00	00	00	4F	FC	0A	FF	FB	00	00	03	15F
B750	FF	FB	00	7F	FC	00	00	00	3F	FC	0F	AC	00	00	00	03	16
B760	FF	FB	00	00	0F	F8	00	00	FF	FC	0F	EB	00	00	00	00	19
B770	FF	F0	00	03	F1	80	00	00	7F	FB	00	03	FF	80	00	00	15C
B780	3F	FB	00	03	FC	00	00	00	7F	FC	00	07	FB	00	00	00	1B
B790	FF	FB	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1BD
B7A0	FF	00	00	3F	80	00	00	00	7F	FC	00	00	00	00	00	00	1BB
B7B0	07	FF	F0	00	00	0F	FF	FC	00	00	00	00	00	00	00	00	100
B7C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B7D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B7E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B7F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B7FF	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
Sun.	A1	BC	E6	26	07	85	EE	A1	66	D6	91	B2	13	3D	FF	95	1B7

	add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
B800	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B810	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B820	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B830	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B840	04	00	02	00	00	00	00	00	00	00	00	01	00	00	00	00	00	106
B850	00	00	00	01	00	00	00	00	1A	00	00	00	00	00	00	00	00	106
B860	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	106
B870	05	00	07	48	68	06	7B	0C	1A	43	00	01	0B	23	5C	94	5E	17A
B880	31	84	18	10	20	28	00	04	B2	00	7C	3F	00	17	20	1F	1F1	
B890	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B8A0	FF	07	0E	07	03	05	02	07	7F	1F	00	F8	03	0C	7C	1F	1C3	
B8B0	0C	FC	0F	0C	3E	07	0E	FF	03	E0	3F	00	0B	3F	03	0F	FCB	
B8C0	3F	C0	FE	07	07	7E	07	7B	C0	00	00	00	00	00	00	00	00	100
B8D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
B8E0	1F	BF	07	7B	3C	61	F0	F0	01	E3	C1	C2	07	0E	FC	7C	10D	
B8F0	00	FB	F8	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
Sum	87	71	84	B4	0F	45	92	19	32	91	FE	05	50	26	D9	4C	CD	177

[illegible]

Bdd	00	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15u0
B800	70	70	7F	FF	05	00	0F	0B	07	0F	07	00	00	00	00	00	00
B810	C0	70	00	07	F1	0E	00	0F	0C	07	00	00	3F	00	00	00	00
B820	3F	00	00	00	01	FE	07	0F	0E	03	00	00	00	00	00	00	00
B830	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B840	E0	00	00	07	F8	00	01	0F	FE	00	00	3F	00	00	00	0F	1B
B850	FF	00	00	07	0B	00	00	00	FF	FE	00	00	0F	0C	00	00	00
B860	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B870	00	00	00	00	FF	0E	00	00	FF	00	00	00	1F	FF	00	00	17C
B880	FE	00	00	03	0F	00	00	00	00	FE	00	00	00	01	FF	00	17E
B890	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B8A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B8B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B8C0	FF	FF	00	00	7F	FF	FF	FF	00	00	00	00	FF	FF	FF	00	13F
B8D0	00	00	00	03	FF	00	00	00	00	00	00	00	00	00	00	00	00
B8E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B8F0	FE	03	00	00	3F	00	00	00	FF	00	FF	00	00	00	00	00	1AD

[illegible][illegible]

音声データ

```

E750 FF 80 01 FF FB C0 00 01 FF FC 00 07 FF BF 00 00 11B
E760 07 FF F0 00 1F FF 7C 00 00 1F FF C0 00 6F FF F2 1CE
E770 00 00 3F FF 80 00 3F BF CB 00 00 FF FE 00 00 37 18B
E780 F9 40 00 07 FF F0 00 03 FF DE 00 00 7F FF 00 00 18B
E790 07 DB CB 00 07 FF FC 00 03 EF F0 00 00 7F FF 00 00 123
E7A0 00 77 DB 00 00 1F FF 80 00 FF F6 00 00 3F FF 00 00 123
E7B0 00 77 DB 00 00 1F FF C0 00 7B F0 00 00 0F FF E0 112
E7C0 C0 01 FF 60 FC 1E 1F B3 01 B3 E3 FB F0 FF 00 00 123
E7D0 7F 80 01 FF E0 1F B0 00 FF E0 00 00 00 1F FF 00 19B
E7E0 00 FF FB 00 71 FB 3F FC 00 03 FF E0 00 FF F0 3E 1AB
E7F0 F0 00 07 FF C0 00 1F F0 FF 80 1F FF 00 01 FE 161

```

Sun 04 D0 0B 16 40 45 24 50 D3 21 B6 D3 60 CB 9C 60 18F

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
E800 03 FC 00 00 7F FC 00 0F FF 87 E0 00 01 FF 3B 07 12E
E810 FF C0 7F 00 00 0F FF 00 01 FF FF E0 00 01 FF E0 10B
E820 00 7F FF C0 00 00 7F FC 00 07 F9 FF 00 00 0F FF 1C2
E830 B0 00 FE 07 F0 00 00 7F 00 00 0F FF E0 00 00 FF 1E1
E840 F0 00 1F FF E0 00 0F FF FB 00 0F FF F0 00 00 7F 162
E850 FB 00 07 FF F0 00 00 7F BC 00 01 FF FB 00 00 3E 160
E860 06 D0 07 FB 03 01 E0 07 FB 07 7C 00 00 00 00 00 1FB
E870 00 03 FF 7F B0 01 FF 00 7F FF FB 00 0F FB 00 7F 1FD
E880 C0 3F D0 07 FC 00 FF 20 0F FF C0 03 FF B0 1F 12D
E890 F0 07 FF FC B0 3F FC 00 FF 00 1F FF EC 01 FF C0 17A
E8A0 3F F0 01 FF FF 60 1F FC 01 FF B0 1F E0 7F 00 7F 1FA
E8B0 F0 07 FC 01 FC 03 DB 07 FF 00 7F B0 1F C0 3F B0 1AE
E8C0 3F C0 07 FB 01 FC 01 FC 03 FC 00 7F B0 3F C0 3F 134
E8D0 B0 3F C0 0F 00 01 FC 03 FC 03 FC 01 FF 00 3F B0 13B
E8E0 1F C0 3F C0 0F 00 03 FC 03 FE 01 FE 00 7F 00 1F 17A
E8F0 00 1F E0 1F C0 1F F0 07 E0 0F B0 0F 07 FE 01 19B

```

Sun FD 29 87 5A 04 87 00 13 CC AD F5 C7 35 FC 01 DF 11B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
E900 DB 1F FF B0 7F B0 FF FB 0F B3 B0 3C 00 0F E0 07 18D
E910 FF FC 01 B7 E0 00 07 F0 FF B0 00 7F B0 03 FF B0 15A
E920 00 7B 07 FB 00 00 1F C0 01 FF 00 01 FF B0 00 0F 192
E930 00 FF FC 00 0F FB 00 1F F0 07 FF C0 00 3F C0 1F 1F5
E940 FE 00 1F FE 00 31 FC 00 FF E0 00 7F 07 BF E0 10C
E950 1F FE 00 FF FB B0 7C 7E 01 FF F0 1F 0F FC 01 E3 193
E960 F0 1F FF 00 FF FF 00 3F FC 03 E0 E0 1F 9F E0 0F 187
E970 FF 00 FB 1B 07 FF E0 3F 1F 01 FF 00 1F B3 C1 E0 196
E980 3E 07 01 E0 00 0F FB 07 C3 C0 7F FC 01 FB 00 1F 1AA
E990 B0 00 00 0F 0F E1 E0 0F 00 FC 00 00 00 00 00 00 15B
E9A0 07 B0 E1 FB 2F E1 40 FF 01 FB 70 03 E0 BF 00 07 1C1
E9B0 FF E0 3E 00 1F FF B0 C0 B0 BC 1F C3 E0 1C 1B F1 19E
E9C0 C1 C1 A7 B0 FB B3 B0 70 76 0F B3 C1 CC 1F F0 111
E9D0 7E 07 C0 30 7F B3 B6 3C 61 E1 C0 1C 0F C3 F1 0F 129
E9E0 1C 1B E0 E0 0E 0F 7C B0 7C 01 E0 7B 3F 00 FF 14F
E9F0 01 FB 1F B1 E0 3F 07 FC 03 F0 7F B7 C0 7C 07 FC 1F5

```

Sun 05 DE 9F C4 2D 1F A1 91 B2 23 A7 C2 B5 13 00 65 1FF

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EA00 0F E0 FB 0F 00 FB 03 FB 1F B1 F0 1F 01 FC 0F FB 19C
EA10 1F B1 FB 3E 03 E0 07 F0 3F B1 E0 3E 07 E0 1F C0 154
EA20 0F 0F A0 FB 0F C0 FE 0F E0 70 07 B0 7C 0F C1 FE 1A2
EA30 0F B0 FB 0F C0 FC 1F E0 FC 07 C0 3E 03 F0 3F B1 105
EA40 0D C6 31 CC 99 99 31 26 26 30 0C B0 66 20 1F 1AC
EA50 FB B0 00 6F F9 66 09 10 6C BB 00 1F FE 00 DE EC 162
EA60 00 00 03 FC 00 3F FF F0 00 00 FF F0 01 FF 3E C0 11E
EA70 00 1F FF B0 03 FF E0 00 00 3F FE 00 07 FF FE 00 1D0
EA80 00 7F C0 00 1F FF 60 00 01 FF FC 00 07 FF 7B 00 173
EA90 01 FF FB 00 7F FF 69 00 01 FF FB 00 07 FF EB 00 1B5
EAA0 01 FF F0 00 07 FE B0 00 03 FE F0 00 07 FB C0 00 179
EAB0 0F FF E0 00 1F F0 00 00 3F FF 00 00 7F FE B0 00 1EB
EAC0 7F FE 00 00 7F F0 00 00 FF FB 00 03 FF C0 00 07 1AC
EAD0 FF 00 00 00 00 00 00 00 00 3F E0 00 05 FF F0 00 110
EAE0 06 70 F0 50 7F B0 03 FB 00 73 00 1F 7E 01 B3 C1 115
EAF0 00 00 0F FF 00 00 7F FF 00 00 00 63 FE 00 07 FF 1E3

```

Sun B5 77 7E 3A 1C 6C 72 FF 0F 0F BB BB 1F F7 B3 C9 1AD

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EB00 C0 07 FF FF 00 01 FF C0 03 FF 9F C0 00 FF F0 00 105
EB10 7F FF F0 00 1F FF 00 07 FF FF B0 00 FF FB 00 7E 1B7
EB20 FF FC 00 07 FF C0 01 FF F3 E0 00 3F FE 00 0F FF 10F
EB30 9F B0 00 FF FB 00 7F FE 3F 00 01 FF E0 00 FF FC 16D
EB40 7C 00 07 FF C0 01 FF F0 FB 00 0F FF B0 03 FF E1 19B
EB50 FB 00 07 FF B0 01 FF E1 FB 00 07 FF B0 01 F1 E1 1B0
EB60 FB 00 07 FF B0 01 FB 7B FC 00 03 FF E0 00 3F FC 10B
EB70 7E 00 01 FF F0 00 0F FF FE 00 00 FF F0 00 07 FF 16F
EB80 01 A7 01 EF C0 65 FC E0 9B B0 10 06 7E 00 7F CC 1BC
EB90 0F E0 00 70 0C FC EC 01 FF 20 03 FF C0 3F C0 134
EBA0 07 FF 00 3F F0 03 FF B0 3F E0 01 FF 00 EE 7F 00 14X
EBB0 7F C0 07 FB 00 7F FE 00 3F C0 3F EB 03 FF FC E0 192
EBC0 0F E0 01 FF 00 3F F3 F0 0F F0 03 FC 00 3F C0 FC 10A
EBD0 01 FC 01 FF 00 0F F0 7F B0 3F B0 3F C0 03 F0 0F 1BB
EBE0 E0 0F 07 FB 00 FE 01 FB 07 FB 03 FC 00 76 00 10E
EBF0 EC 01 F6 01 E0 1D C0 EC 03 FE 01 FE 00 7F 60 18A

```

Sun 39 B0 F5 2D DC 04 37 B8 AB 66 22 29 E7 EA AA 0E 1BC

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
EC00 FE 01 FB 00 FF B0 0F E0 7D 00 7F C0 1F C0 03 FC 102
EC10 07 F0 07 F0 00 3B 00 7F F0 03 FB 01 FE 00 07 C0 156
EC20 03 C3 C1 C0 3F FC 0F 07 C0 7C 0E 1F B0 7F F0 0E 1F1
EC30 0F F0 01 FF FB 00 00 03 FF FF B0 F0 00 00 3F 197
EC40 B1 FC 01 F9 FB 07 FF 01 FC FC 01 F1 F0 1F 9F 03 15F
EC50 FF FC 01 FB F0 1F FF 01 FB 7E 00 FB FB 0F E7 00 18B
EC60 FF 3F 00 FB 7C 07 E1 B0 FE 1F B0 7C 3C 07 E0 C0 116
EC70 7F 00 7C 3E 07 E0 C0 7E 1F FB 0B 3C 0F C0 E0 16F
EC80 7C 3F 01 FB 7B 1F C0 E0 FC 3E 03 F0 70 3F B0 0F 137
EC90 7F 0F 00 03 FB 00 3F 03 F0 0E 0F B0 FD 03 EF 176
ECA0 C0 7E 00 03 FB 0F FF 00 FC 07 03 E0 B0 07 E0 19E
ECB0 3F B0 01 FC 03 FF 00 1F C0 00 FC 00 FC 00 00 195

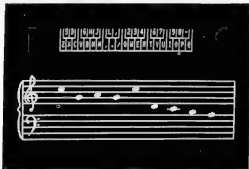
```

Sun 0F EB C7 0B 4E 0D 9C E6 5B 6B 75 9C DC 2E AA 68 1BC



MZ-80B

MUSICREATE



●柴田文彦

MZ-80Bには音楽演奏機能があり、プログラム中で使うと効果的なのですが、そのデータを作るのが少々やっかいです。そこで楽しみながら簡単にシャープ・フォーマットの楽譜?が作れてしまうソフト、名付けて“MUSICREATE”を作りました。

♪ プログラムの使い方

プログラムはすべてマシン語で書かれています。モニタなどで入力してください。スタート番地は\$8000です。

スタートさせると、楽譜を表示しながらテーマ曲を演奏します。テーマが終わると次はモードの選択をします。モード①では、キーボードが鍵盤になり演奏できます。演奏が終わったら[BREAK]キーを押してください。つぎに[P]キーを押すと、いまあなたが演奏した通りに演奏し返してきます。つぎに[M]キーを押せば、またあなたが演奏することになります。やめたいときは[R]キーを押せばモード選択に戻ります。キーボードのキーと鍵盤との対応は画面に表示されるので、そちらを見てください。

モード②は楽譜を作ったり、シャープ・フォーマットでプリントしたりするためのものです。メニューが1～8まであります。

メニュー1は、楽譜のもとになる音符を入力するものです。モード③のとく同様にして入力してください。間違えたら[DEL]キーで1音消すことができます。

メニュー2は入力した音符を確めるために、ワンキープレイをするものです。表示される楽譜を見ながら[SPACE]キーを使って演奏してください。ここで思い通りに音符が入力されていたら、つぎはメニュー4に移ります。

メニュー4ではメニュー1で入力した音符に音長を付け加え、必要に応じて休符を入れて完全な楽譜を作ります。音符が表示されたら、その音符に付けたい音長を0～9の数字で入力します。この数字と音長との対応は表1を見てください。前の音長と同じ場合には[CR]のみで結構です。休符を入れた場合には[+] [CR]を押してください。このときも休符の長さを0～9で入力します。すべて終了したら、メニュー3を選んで音符を実際に演奏させてみましょう。

以上の操作で楽譜はできました。メニュー5を選んでシャープ・フォーマットの楽譜を見てみましょう。プリントし出力する場合は[P]を入力してください。つぎにASCIIかHEXか聞いてきます。楽譜をBASICで使う場合には

[A]を入力し、マシン語プログラム中で用いる場合には[H]を入力するとよいでしょう。

でき上がった音楽データは、メニュー6、7によってテープにSAVE、LOADが可能です。1度SAVEしたデータは、LOADすれば再び演奏やプリント・アウトが可能です。

メニュー8を選べば、モード選択に戻ります。

BASIC中で音楽データを用いる場合、メニュー6でSAVEしたテープからストリング変数に代入する方法を紹介しますが、モニタでテープをLOADしBASICに戻ってからつぎのようなプログラムを走らせてください。

```
10 LIMIT$=FFFF
20 D=49152
30 A=PEEK(D)
40 IF A=13 THEN END
50 M$=M$+CHR$(A)
60 D=D+1:GOTO 30
```

以上のプログラムを走らせると、M\$にデータが代入されます。ただし、データはの場合255文字までです。この後はMUSICで演奏してください。

以上簡単な説明で判り難い点もあったかと思いますが、使ってみれば解っていたでしょう。

♪ プログラムについて

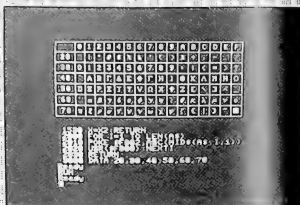
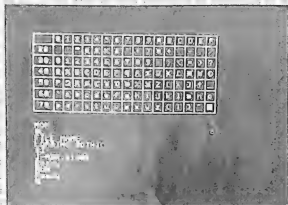
このプログラムはいつものようにBASEで書きました。BASEで書くときは紙に書かなくてもどんどん打ち込めるので、結局わりによくいプログラムになってしまいがちです。しかし、各処理はまとまったブロックになっています。表2に主要ルーチンのアドレスを示します。

タイトルでは楽譜を表示しながら演奏しますが、このテーマ曲を変えたいときは\$9086からのデータを変更してください。データのフォーマットは音程番号、音長番号、音程番号、音長番号……の繰り返しで、エンド・マークは\$FFです。ここで音程番号と実際の音程との対応を表3に示します。音長番号は表1と同じです。また、楽譜を表示しながら演奏するルーチンは\$8AB9からスタートします。上記フォーマットで書いたデータの先頭アドレスをDEレジスタに入れ、五線譜を描くルーチン\$8829を呼んでから\$8AB9を呼んでください。

グラフィック・ キャラクタ・ジェネレータ

★グラフィック画面にキャラクタ表示を！

●柴田文彦



MZ-80Bでグラフィックとテキスト画面を同時にプリンタに出すには特別なプログラムが必要です。またグラフィックで表やグラフを描いて文字を入れる場合、1キャラクタ単位ではグラフの目盛りとずれてしまう場合もあるでしょう。それでこれらの問題を解決するのがこのプログラムです。

1 プログラムの機能

このプログラムはG-RAMに1ドット単位で位置を指定して任意のキャラクタを描くものです。機能的にはBASISICのPATTERN文と似ていますが、使いやすさはかなり上だと思います。

キャラクタ・フォントはRAM上に持っているいろいろなフォントが使えます。今回は標準ASCII体以外に8種のフォントを用意しました。表1～9を見てください。このプログラムでは8種のうち1種をメモリにロードして、ASCII体と切り換えて使うようになっています。

2 使用方法

まず、プログラム本体と標準ASCII体をF800～FCFFにロードしてください。次に、8種のフォントのうち1つを選び、これをFD00～FDFFに入れます。これで準備完了です。

文字を表示させるには、32(\$20)～127(\$79)までのコードを\$F802に入れて、\$F800をコールします。\$F802に31以下の数を入れるとコントロール・コードとなり、数値に対応した動作をします。これは表10のコントロール・コード表を見てください。

文字の位置を決めるには、X座標下位バイトを\$F803

に、上位バイトを\$F804に、またY座標を\$F805に入れておきます。ただし1文字プリントすることに座標は自動的に右に1文字分ずれるので、横に連続した文字をプリントするときには初めの1文字のみ座標指定してください。なおこのカーソルは右端に行くと次の行の先頭に自動的に移動し、下端に行くとグラフィック画面全体が1行分スクロールします。

以上の説明では、初心者の方には理解しにくいと思うので、具体的な例をあげて説明します。まずBASICプログラム中の変数を以下のように定めます。

C: プリントする文字コード、またはコントロール・コード
X: 文字位置 (カーソル) のX座標 (0～319)
Y: 文字位置 (カーソル) のY座標 (0～199)

以上の変数に目的の値を代入して以下のサブルーチンを呼びます。

```
1000 POKE$F802,C
1010 X1=0
1020 IF X1=256 THEN X=X-256:X1=1
1030 POKE$F803,X:POKE$F804,X1
1040 POKE$F805,Y
1050 USR($F800):RETURN
```

これでOKです。

3 蛇足

これは実は重要なことですが、BASICから使うときには、LIMIT\$F7FFをプログラムの先頭に入れておいてください。このプログラムはBASICから使うことを前提に作ったので、Z80のレジスタの保存はしていません。IX、IY以外はすべて破壊されます。機械語プログラマーの方は注意してください。

表1 ASCII体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	P	R	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	O	R	S	T	U	V	W	X	Y	Z	<	=	>	?	
60	"	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	<	=	>	?	

表3 COUNT体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	P	R	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	O	R	S	T	U	V	W	X	Y	Z	<	=	>	?	
60	"	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	<	=	>	?	

表5 GREEK体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο	Π
50	ρ	σ	τ	υ	φ	χ	ψ	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η
60	θ	ι	κ	λ	μ	ν	ξ	ο	π	ρ	σ	τ	υ	φ	χ	ψ
70	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο

表7 OUTLINE

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο	Π
50	ρ	σ	τ	υ	φ	χ	ψ	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η
60	θ	ι	κ	λ	μ	ν	ξ	ο	π	ρ	σ	τ	υ	φ	χ	ψ
70	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο

表9 SLANT体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο	Π
50	ρ	σ	τ	υ	φ	χ	ψ	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η
60	θ	ι	κ	λ	μ	ν	ξ	ο	π	ρ	σ	τ	υ	φ	χ	ψ
70	ω	Ω	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Κ	Λ	Μ	Ν	Ξ	Ο

図1 メモリ・マップ

F800

FA00

FD00

FFFF

本 体
フォント1
(ASCII体)
フォント2

表2 BYTE体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	O	R	S	T	U	V	W	X	Y	Z	<	=	>	?	
60	"	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	<	=	>	?	

表4 GOTHIC体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	O	R	S	T	U	V	W	X	Y	Z	<	=	>	?	
60	"	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	<	=	>	?	

表6 カタカナ

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	"	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
40	"	タ	チ	ツ	テ	ト	ナ	ニ	ノ	ハ	ヒ	フ	ヘ	ホ	マ	ミ
50	"	ム	メ	モ	ヤ	ユ	ヨ	リ	ロ	ワ	ヰ	ヱ	ヰ	ヱ	ヰ	ヱ
60	"	ヲ	ツ	チ	テ	ト	ナ	ニ	ノ	ハ	ヒ	フ	ヘ	ホ	マ	ミ
70	"	ム	メ	モ	ヤ	ユ	ヨ	リ	ロ	ワ	ヰ	ヱ	ヰ	ヱ	ヰ	ヱ

表8 ROMAN体

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	"	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	O	R	S	T	U	V	W	X	Y	Z	<	=	>	?	
60	"	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	<	=	>	?	

表10 コントロール・コード

コード(16進)	機 能
01	カーソル1文字分左に移動
02	カーソル1文字分上に移動
03	カーソル1文字分右に移動
04	カーソル1文字分下に移動
05	カーソルを0、0に移動
06	グラフィック画面のクリア
0A	ライン・フィード
0D	キャリッジ・リターン (LFなし)
0E	キャラクタ・セットをフォント1に
0F	キャラクタ・セットをフォント2に

コード表印字BASICプログラム

```

10 LIMIT $F7FF
20 GRAPH C,11,01
30 FOR Y=0 TO 105 STEP 15
40 LINE 0,Y,269,Y
50 NEXT Y
60 FOR X=30 TO 270 STEP 15
70 LINE X,0,X,105
80 NEXT X
90 LINE 0,0,0,105
95 C=48:POKE $F80C,0
100 FOR X=34 TO 259 STEP 15
110 Y=41:GOSUB 1500
120 POKE $F802,C:USR1($F800)
130 C=C+1:IF C=58 THEN C=65
140 NEXT X
150 FOR Y=19 TO 94 STEP 15
160 X=41:GOSUB 1500
170 READ A$:GOSUB 2000

```

```

180 NEXT Y
185 C=32:POKE $F80C,1
190 FOR LN=1 TO 6:FOR CL=0 TO 15
200 X=CL*15+341+LN*15+4:GOSUB 1500
210 POKE $F802,C:USR1($F800):C=C+1
220 NEXT CL:LN
230 END
1500 X2=X*11+0:IF X*255 THEN X1=11*X-256
1510 POKE $F803,X1:POKE $F804,X1
1520 POKE $F805,Y
1530 X=X2:RETURN
2000 FOR I=1 TO LEN(A$)
2010 POKE $F802,ASC(MID$(A$,I,1))
2020 USR1($F800):NEXT I
2030 RETURN
2000 DATA 20,30,40,50,60,70

```

マシン語ダンプ・リスト (ASC II 体データ含む)

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FB00 18 0B 00 00 00 00 FF FF 00 00 0A 03 0B 155
FB10 47 7A 05 0B FB 8B 20 0C 3A 05 FB 47 3A 0B FB 8B 20 0C 3A
FB20 02 1B 03 0B 3B FB 0D 47 FB 09 21 0B 5E 11 3B 00 00 16A
FB30 7A 05 FB 3C 47 19 10 FB E5 2A 03 B1 08 00 01 104
FB40 00 00 B7 02 52 04 7C B5 87 2B 0C 7E FF 20 F2 1A1
FB50 AF 95 AF E 08 95 05 02 08 0F E1 16 00 5F 19 22 152
FB60 09 FB 09 FB 0C 16 FA 02 FB 0E 20 0A 2C 1F FB 80 175
FB70 3A 4B F5 5C 0C FB 21 FB 09 B7 2B 03 05 FB FC 1 1AB
FB80 11 0B 00 0B 47 47 19 10 FD 5A 2A 09 FB 06 0B 165
FB90 C5 E5 3A 0B FB 47 0E 0A 1A 13 04 05 08 06 B7 17 16E
FBAA CB 11 10 FA B6 77 27 7B 86 77 E1 05 11 2B 00 19 1E4
FBAB D1 C1 10 CB 01 FB 7A 05 FB 32 0B FB 0D 23 1F 156
FBBC 09 2A 05 FB 11 0B 00 19 22 05 FB 24 25 2B 71 7D 14C
FBD0 FE 39 3B 1C 21 00 00 02 07 FB 3A 05 FB C6 08 32 100
FBE0 05 FB FE C1 30 01 09 D6 08 32 05 FB 0C FB FB C9 14C
FBF0 22 0A FB 2A 09 FB 23 22 09 FB 09 11 00 60 21 40 12C

```

```

Sum E 5A 70 E9 AA B2 F2 70 9C BE 18 B7 E7 CF 3B 87 1BB

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
F800 61 01 00 1E ED 00 10 7E 01 40 01 AF 77 23 0E 152
F810 7B 81 C2 0C F9 09 DB EB C7 F7 CB FF D3 EB 7E 02 103
F820 D1 F4 C9 DB EB C7 CB BF 07 ED C9 FE 01 2B 26 130
F830 FE 02 2B 27 0E 28 31 FE 04 2B 32 FE 08 28 45 175
F840 FE 3B AB FE 0A 2B 08 00 2B 5E 0E 3B 5A 1C 2E 16A
F850 FE 0F 0B 3C 1B 5F 0A DB FB 5A 3A 05 FB FE 1B 156
F860 7B 57 0B 08 32 05 FE 1B 4C 0C C1 FB 18 47 2A 03 10E
F870 7B 70 2A 25 20 0A FE 0B 3B 3B 11 0B 00 0B ED 52 16A
F880 22 05 FB 18 50 21 00 00 23 FB AF 42 05 0B 10 199
F890 2A AF 12 00 60 11 0B 60 11 00 60 01 3E 1F 0B 156
F8A0 1B 17 21 00 00 22 05 FB 1B 0B AF 32 0C FB 1B 05 1BE
F8B0 7E FF 32 0C 0B CD 23 F9 C9 00 FF 00 FF 00 FF 00 122
F8C0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 122
F8D0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 122
F8E0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 122
F8F0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 122

```

```

Sum E 51 76 26 B8 DA E9 3D 0A 06 71 6D 11 B5 BE FC 117

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FA00 33 00 00 00 00 00 00 00 0B 0B 0B 0B 00 00 00 130
FA10 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14
FA20 3C 0A 1C 2B 1E 0B 0A 06 2A 10 0B 04 3E 00 00 140
FA30 0A 0A 0A 0A 2A 12 2C 00 0B 0B 0B 00 00 00 00 19C
FA40 0B 0A 02 02 02 0A 0B 0B 0B 10 20 20 10 0B 00 1AE
FA50 0B 2A 1C 1B 1C 2A 08 0B 0B 0B 0B 0B 0B 00 00 102
FA60 0B 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 1A2
FA70 0B 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 15A
FA80 1C 22 22 2A 2A 2C 1C 00 0B 0C 0B 0B 0B 0B 0B 14E

```

```

FA90 1C 22 2A 7A 1A 02 7E 00 1C 22 2A 7E 00 1B 20 22 1C 00 1BE
FAA0 1B 18 14 1E 1E 10 10 10 10 7E 02 1E 20 20 22 1C 00 1BB
FAB0 1B 04 02 1E 22 22 1C 1C 00 7E 20 10 0B 08 04 0A 00 122
FAC0 1C 22 1C 22 1C 22 1C 1C 00 1C 22 22 7E 20 1A 0E 00 1B6
FAD0 00 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0A 16E
FAE0 10 0B 0A 02 0A 0B 1A 00 00 06 0E 0E 0E 0E 0E 0E 0E 1B6
FAF0 04 0A 10 20 10 10 0B 0A 1C 22 1E 0B 0B 00 0B 0E 1BE

```

```

Sum C0 2A F0 DA 4B FE 0F 10 0A 22 68 5A 7B CC CA 0A 1CE

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FB00 1C 22 2A 7A 1A 02 7E 00 1C 22 2A 7E 00 1B 20 22 1C 00 1BE
FB10 1E 22 2E 2E 2E 2E 2E 1E 0E 02 02 0E 02 0E 02 0E 1B6
FB20 1E 22 2E 2E 2E 2E 2E 1E 0E 02 02 0E 02 0E 02 0E 1B6
FB30 2E 02 0E 1E 02 02 0E 02 0E 02 0E 02 0E 02 0E 0E 1B6
FB40 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 1B6
FB50 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 1B6
FB60 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 1B6
FB70 22 2E 2A 2A 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 1B6
FB80 1E 22 1E 02 02 0E 0E 1C 22 2E 2E 2E 2E 2E 2E 2E 2E 1B6
FB90 1E 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 1B6
FBA0 7E 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 1B6
FBB0 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 1B6
FBC0 22 22 1A 18 14 22 22 22 22 22 22 22 22 22 22 22 1B6
FBD0 7E 20 10 0B 0A 02 0E 0E 0E 0E 0E 0E 0E 0E 0E 0E 1B6
FBE0 2E 1A 0B 7E 0B 7E 0B 7E 0B 7E 0B 7E 0B 7E 0B 7E 1B6
FBF0 00 00 0E 1A 22 00 00 00 00 00 00 00 00 00 00 00 1B6

```

```

Sum 1A 92 7C EA 4F 42 B4 00 7A 0C 3B 66 9E 90 3E 7F 19F

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FC00 0A 0B 10 00 00 00 00 00 00 00 00 1C 20 3C 22 3C 00 1F2
FC10 02 02 1E 22 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 124
FC20 20 3C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 108
FC30 1B 2A 04 1E 0A 04 0A 0A 0A 0A 0A 0A 0A 0A 0A 144
FC40 02 02 1E 22 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 102
FC50 10 0B 10 10 10 10 10 10 10 10 10 10 10 10 10 10 1F0
FC60 0C 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 126
FC70 00 00 1E 22 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 144
FC80 00 00 1E 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 1B0
FC90 00 00 3A 0E 02 02 0E 0E 0E 0E 0E 0E 0E 0E 0E 0E 1DE
FCBA 0A 1E 0A 0A 2A 1B 0B 00 00 22 22 2E 2E 2E 2E 2E 152
FCBB 00 22 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 150
FCBC 00 2E 1A 0B 1A 22 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 142
FCBD 00 0E 1E 0B 0A 3E 0B 3B 0C 0C 0A 0C 0A 0C 0B 0E 17E
FCBE 0B 0B 0B 0B 0B 0B 0B 0B 0B 0B 1B 1B 10 1B 0E 0E 1EC
FCBF 2A 1A 0B 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 1F0

```

```

Sum 9A 7E CA 3B 0E 1C 5C 16 CF A5 01 0B 4F 5F 05 57 192

```

BYTEデータ

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FD00 00 00 00 00 00 00 00 00 1C 1C 1C 1C 1C 1C 1C 1A8
FD10 56 2A 12 00 00 00 00 00 22 7F 22 22 7F 22 00 14A
FD20 0B 7E 02 3E 20 3E 0B 00 47 25 17 0B 7A 52 71 0AE
FD30 0E 12 02 0A 18 12 0E 1C 1C 10 0C 00 00 00 00 1C2
FD40 1B 0C 0A 06 0A 0C 1B 0A 0A 0B 0B 0B 0B 0B 0B 1AE
FD50 0A 15 0E 0A 0E 1C 10 0A 00 0B 0B 3E 1B 1B 00 1D0
FD60 00 00 00 00 00 1C 1C 1C 0C 00 00 3E 3E 00 00 1D0
FD70 00 00 00 00 1C 1C 1C 0A 20 10 0B 04 02 01 00 1D3
FD80 1F 11 19 15 13 1F 07 04 04 0A 04 0A 1F 0D 1B0
FD90 3F 20 20 3F 03 03 3F 1E 10 1E 1B 18 1E 1AE
FDA0 11 11 11 3F 1B 0B 1F 01 01 1F 1B 19 1F 0E 1A3
FDB0 1F 11 01 1F 17 17 1F 1F 11 0B 0A 06 0A 0E 1A3
FDC0 1F 11 1F 17 17 1F 1F 1F 11 1F 1B 19 1F 0E 155
FDD0 00 1C 1C 1C 1C 1C 00 00 1C 1C 1C 1C 1C 1C 1C 1FC
FDE0 10 0B 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 15A
FDF0 01 02 0A 1B 0A 02 01 00 7E 42 40 7B 0B 0B 0A 1AE

```

```

Sum 24 31 BC 1C 15 25 43 0C E7 E3 5D CA DB A5 50 0C 17E

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FE00 3C 24 3C 00 00 00 00 1F 11 3F 21 23 23 23 00 195
FE10 1F 11 3F 21 23 23 23 00 3F 21 01 03 03 03 3F 1DE
FE20 1F 21 21 23 23 23 1F 00 3F 01 0F 02 03 0F 0E 17E
FE30 01 01 01 03 03 03 03 3F 21 01 09 23 23 3F 0B8
FE40 1B 01 21 21 23 23 23 0A 04 0C 0C 0C 0C 0C 0C 147
FE50 20 20 20 30 30 31 1E 00 21 11 09 0F 17 23 23 1B2
FE60 01 01 01 03 03 03 03 3F 49 49 49 49 49 49 49 146
FE70 27 29 29 29 29 29 29 3F 21 21 21 21 21 21 21 152
FE80 27 21 21 21 21 21 21 03 3F 21 21 21 21 21 21 152
FE90 2F 21 21 21 21 21 21 03 3F 01 3F 3F 3F 3F 3F 158

```

COUNT体データ

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FDB0 00 00 06 06 06 06 06 06 0C 0C 0C 0C 0C 0C 0C 148
FDB1 26 24 12 00 00 00 00 16 7F 7F 16 7F 16 7F 16 1B1
FDB2 0E 1E 27 09 1E 0C 00 00 21 0B 04 52 71 0B 165
FDB3 1E 27 07 77 77 77 77 0C 0A 06 06 06 06 06 06 1EE
FDB4 1B 1C 1E 0E 0E 1C 1B 0B 0E 0E 1C 1B 1C 1E 0E 192
FDB5 0C 2B 1E 1E 1E 2B 0C 00 0C 1E 1E 0C 0E 0E 141
FDB6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 141
FDB7 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1B8
FDB8 7E 7F 27 27 27 27 1C 1C 1C 1C 1C 1C 1C 1C 1C 15F
FDB9 0E 1E 1E 0E 0E 0E 0E 0E 7F 7F 7F 7F 7F 7F 7F 161
FDBA 1C 1C 1E 1E 1E 1E 1C 1C 1C 1C 1C 1C 1C 1C 1C 120
FDBB 1E 27 77 1F 27 27 1E 0E 7F 1C 1E 0E 0E 07 07 17B

```

```

FDB0 1E 27 27 1E 27 27 1E 00 1E 39 39 3B 39 1E 00 153
FDB1 00 06 06 06 06 06 06 06 00 06 06 06 06 06 06 150
FDB2 39 39 39 39 39 39 39 39 06 06 06 06 06 06 06 156
FDB3 02 07 1C 0C 0E 07 03 1E 39 3B 3B 1C 00 1C 00 157

```

```

Sum 5B B1 39 15 B6 B4 0F 6B CB C0 C5 BB AF 35 02 1DD

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
FE00 1E 25 2B 1B 03 7E 1E 1F 39 39 3F 39 39 39 39 17F
FE10 1F 27 27 1F 27 27 1F 0E 1E 27 07 07 07 07 07 190
FE20 1F 27 27 27 27 27 1F 0E 7E 07 07 07 07 07 07 188
FE30 7E 07 07 1F 07 07 07 0E 1E 27 07 07 07 07 07 14F

```


Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum
FE00	18	24	18	00	00	00	00	00	3B	44	44	3E	22	11	11	00	:96
FE10	3B	44	64	1E	22	11	1F	00	3B	44	04	02	02	21	1F	00	:14
FE20	3B	24	44	42	42	31	0F	00	7B	04	04	1E	02	01	1F	00	:24
FE30	7B	04	04	1E	02	01	01	00	3B	44	04	02	32	21	1E	00	:95
FE40	44	44	22	3E	22	11	11	00	7E	10	0B	08	04	04	3F	00	:11
FE50	70	40	20	20	12	11	0F	00	0B	4B	24	12	0E	11	21	00	:EB
FE60	0B	04	04	02	02	01	1F	00	3B	54	52	42	22	11	11	00	:9B
FE70	4C	4C	4A	2A	29	19	19	00	3B	44	44	22	22	11	0E	00	:BA
FE80	3B	44	44	3E	02	01	01	00	3B	44	44	22	2A	31	5E	00	:9D
FE90	3B	44	44	3E	0A	09	11	00	3B	44	02	1C	20	11	0E	00	:FB
FEA0	7E	0B	0B	04	04	02	02	00	44	44	22	22	22	11	0E	00	:A7
FEB0	44	44	22	22	12	0C	00	44	44	22	2A	2A	15	0E	00	:2D	
FE00	42	24	14	0B	14	12	21	00	44	42	22	1C	0B	04	02	00	:9B
FED0	7E	40	60	1B	06	01	3F	00	1E	02	02	02	02	1E	00	:C2	
FE00	01	02	04	0B	10	20	40	00	3C	20	20	20	20	3C	00	:97	
FE00	0B	1C	2A	0B	0B	0B	0B	00	00	00	00	00	00	00	00	:6E	
Sum	03	DA	AD	DA	29	DB	4F	00	AC	34	E0	A6	6E	19	DO	00	:4C

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum
FF00	00	00	00	00	00	00	00	00	00	00	00	00	1C	26	22	3C	:00
FF10	0B	04	04	1E	22	23	1D	00	00	00	00	00	1C	02	22	1C	:00
FF20	40	40	20	3C	22	22	1C	00	00	00	00	00	1C	22	12	3C	:00
FF30	30	4B	0B	1E	04	02	02	00	00	00	00	00	1C	22	22	3C	:11
FF40	0B	04	04	1A	26	21	11	00	00	10	00	0B	0B	04	04	00	:AA
FF50	00	40	00	20	20	10	11	0E	0B	04	04	12	0A	05	09	00	:E9
FF60	10	0B	0B	04	04	02	02	00	00	00	00	00	3E	2A	15	15	:00
FF70	00	00	04	1C	24	12	12	00	00	00	00	00	1C	22	11	0E	:00
FF80	00	00	00	1C	22	22	1D	01	00	00	00	0E	11	11	5E	30	:1C
FF90	00	00	00	1B	24	02	02	00	00	00	3B	44	1B	22	1C	00	:12
FFA0	00	0B	3E	04	02	22	1C	00	00	00	00	00	44	22	1C	00	:2E
FFB0	00	00	00	44	22	14	0B	00	00	00	00	00	2A	15	15	1E	:00
FFC0	00	00	00	22	1C	0E	11	00	00	00	00	00	44	22	3C	10	:0E
FFD0	00	00	06	1B	0B	06	1B	00	00	00	00	00	00	00	00	00	:14
FFE0	00	0B	0B	0B	0B	0B	0B	00	00	00	00	00	00	00	00	00	:3B
FFF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum	90	EB	BB	90	4C	02	E5	17	0B	14	5B	EE	4C	67	B9	4C	:F4



MZ-80B

PATTERN EDITOR



《パターン・エディタ》



■田中 聡

MZ-80Bの「漢字データ作成プログラム」で作った漢字、ひらがな一覧をCRTディスプレイ上に表示した写真を見方も多いと思います。

私もBASICのグラフィック・コントロール文を使った、「パターン・エディタ」なるものを作ってみました。

ただし、このプログラムではV-RAMの1だけを使っています。

プログラムの機能

- ①パターン・コード：Pコマンド 登録されたデータのパターン・コードを指定して、画面上に呼び出す。
- ②セーブ：Sコマンド 主記憶上のデータ・ファイルをCMTにセーブする。
- ③ロード：Lコマンド CMT上のデータ・ファイルを主記憶にロードする。
- ④ディスプレイ：Dコマンド 登録されている全データをCRTディスプレイ上に表示する。
- ⑤クリア：Cコマンド 画面のクリア。
- ⑥エディット：Eコマンド CRTディスプレイ上から16×16ドットでグラフィック・データを作成する。

各コマンドの説明

①パターン・コード

□キーを入力すると、CRTディスプレイ下部に“P.C.ode”が表示され、キー入力待ちになります。

目的とするデータのパターン・コードを10進数で指定し、**[CR]**入力で、画面右上に目的のグラフィック・データが表示され、同時にストリング・データも表示されます。

この機能によって、登録されたデータのストリング・データが1文字単位で参照でき、漢字などを使ったタイトル文を作るときに役立つと思います。

パターン・コードは30～167まで指定可能です。

②セーブ

[S]キーを入力すると、主記憶上に登録されたデータのみがCMTにセーブされます。ただし、最後のデータの後はエンド・コード(999)が付加されています。

③ロード

[L]キーを入力すると、CMTにセーブされているデータを主記憶にロードします。データの終了はエンド・コードで判定しています。

④ディスプレイ

[D]キーを入力すると、全画面消去の後、登録されている全データがCRTディスプレイ上に順に表示されていきます。

表示の終了でモニター・サブルーチン(\$0F14)のBELLを2回鳴らし、次のコマンドの入力を可能にします。

[C]キーの入力で画面は元に戻ります。そのまま続けて登録してください。

⑤クリア

[C]キーを入力すると、画面上の表示データが消去されます。

次のデータを作成するときなどに使いますが、**[C]**キーを使わないで、続けて編集作業(**[E]**コマンド)を始めることもできます。

⑥エディット

[E]キーを入力すると、画面上にカーソルが表示されてくるので、漢字、ひらがななどを16×16のドット・マトリックスで表現してください。

入力キーはスペース以外ならなんでもかまいません。**[CR]**入力すると、5～6秒後、グラフィック・データとストリング・データが表示されてきます。

■補足説明

・各コマンドを実行している間は画面下部に実行中のコマンドが表示され続けています。ただし、**[D]**コマンド実行中は表示されません。

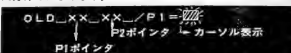
・**[S]**コマンドではデータが32×7バイトの論理レコード長単位で、CMTにセーブされていきます。

よって、**[L]**コマンドでも同様です。そのため、ポインタは7単位おきに進んでいるので、データは7文字単位で作った後、セーブしなければ、途中で空ができてしまいます。

ただし、データ修正用コマンドを使ってポインタを戻すことも可能です。

・データ修正：□コマンド

[%]キーを入力すると、画面下部へ次の表示を出し、入力待ちになります。



P1に修正先のポインタを指定した後、**[CR]**入力で“P2=”が表示され、またキー入力待ちになります。P2のポインタを指定後、**[CR]**入力で編集ポインタが変更されます。

写真1

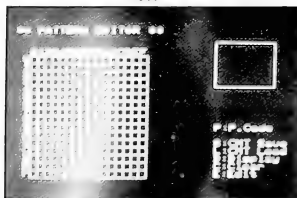


写真2

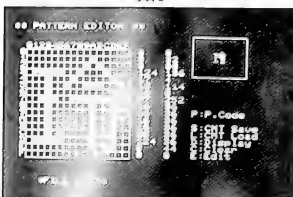


写真3

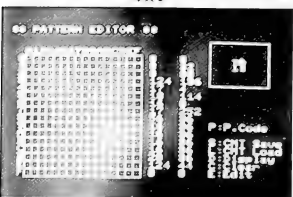


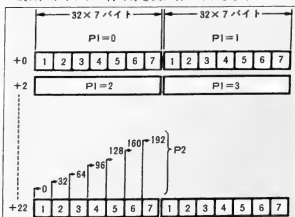
写真4



[E]コマンドでデータを修正した後、P1、P2のポイントを元に戻してください。いままで登録したデータが、すべて表示可能になります。

ただし、本コマンドはデバッグ用に組み込んだ機能であるため、修正されたポイントに対し、ノーチェックですから注意して使ってください。

次に、ポイントの持ち方を図に書いておきます。



P1は0～23、P2は0～192までとします。



PATTERN EDITORプログラム・リスト

```
80 REM *** PATTERN EDITOR ***
90 REM
100 REM C.DATE '81.06.14
110 REM AUTHOR S.TANAKA
115 REM
120 DIM DT(24,224),Wk(32)
990 P1=0:P2=0
1000 REM
1002 REM *** PAGE 1 PROGRAM ***
1004 REM
1010 GRAPH I1,C,01:PRINT CHR$(6)
1020 FOR N=32 TO 160 STEP 8
1030 LINE 24,N,152,N:NEXT
1040 FOR N=24 TO 152 STEP 8
1050 LINE N,32,N,160:NEXT
1060 LINE 224,16,288,16,288,64,224,64,224,16
1070 CURSOR 0,0
1080 PRINT " ** PATTERN EDITOR ***"
1085 CURSOR 3,3
1090 PRINT "0123456789ABCDEF"
1100 FOR X=3 TO 18
```

```
1110 CURSOR 2,X+1
1120 PRINT CHARACTER$(X,3):NEXT
1130 REM
1140 CURSOR 28,13:PRINT "P:P.Code"
1150 CURSOR 28,15:PRINT "S:CMT Save"
1160 CURSOR 28,16:PRINT "L:CMT Load"
1170 CURSOR 28,17:PRINT "D:Display"
1180 CURSOR 28,18:PRINT "C:Clear"
1190 CURSOR 28,19:PRINT "E:Edit"
1200 REM
1202 FOR Y=4 TO 19
1204 CURSOR 3,Y
1206 PRINT SPACE$(25):NEXT
1208 REM
1210 GET IN$
1220 IF IN$="P" THEN 5000
1230 IF IN$="S" THEN 2000
1240 IF IN$="L" THEN 2500
1250 IF IN$="D" THEN 3000
1260 IF IN$="C" THEN 3500
1270 IF IN$="E" THEN 4000
```

図1 MZ-80BP5のビット・データ転送

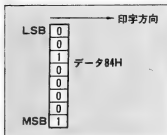


図2 グラフィックRAM

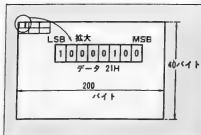


図3 グラフィックRAMデータをそのままプリンタへ出力

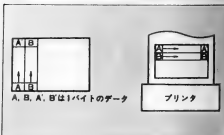
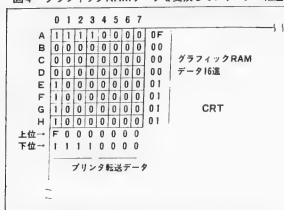


図4 グラフィックRAMデータを変換してプリンタへ転送



合には逆の手を使えば良いのです。つまり、グラフィックを640ドットにすればキャラクタの80文字モードのドットと一致します。

原理は実に簡単にグラフィックの横1ドット分をプリンタ上で横2ドットに表わせればOKです。幸い横2ドットにしてもプリンタを縮小モードにすれば、ほとんど見分けが付きません。

さて、このプリンタ方式ですが、ネックが2つあります。この方法では、キャラクタとグラフィックを合成するバッファがRAM上に必要です。バッファのエリアは最低8バイトから最大16Kバイトが必要です。

16Kバイトのバッファを確保すれば、グラフィックRAM 1とRAM 2のそれぞれ8Kバイトがすべてバッファ上に移るので、プログラミングが非常に楽になります。逆に、バッファを減らせばそれだけプログラミングは複雑になります。このプログラムでは640バイト使っています。

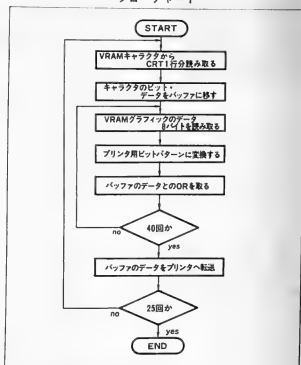
もう1つのネックは、プリンタの出力パターンをどの向きにするかということです。判り易く説明するために、プリンタの通常のモードでキャラクタを出力する場合を(A)、それを90°回転させた場合を(B)とします。

なお、ここで断っておきますが、最近市販されたソフトでキャラクタとグラフィックを同時コピーできるものがあります。このソフトはBに相当しますが、コピー画面がA4判相当になるなど、他の機能がプラスされています。したがって、ここでのA、Bに関する考察はあくまでも私がプログラムを開発する際に検討したものだと考えてください。

さて、AにするかBにするかがなぜ問題になるのでしょうか。これを理解するには、プリンタMZ-80BP5のビット・イメージ・モードとMZ-80B本体のグラフィックRAMのグラフィック・データの記憶方式について触れる必要があります。

なお、この辺のことは、プリンタの解説書、MZ-80Bオーナーズ・マニュアルにはあまり詳しく記載されて

フローチャート



いません。私があればこれ試してみても、たぶんこうであろうと結論したものです。

EPSONのMPシリーズなどでビット・イメージのデータの転送方法についてはかなり知られていますし、詳しい解説書も出ています。MZ80-BP5へのビット・イメージの転送は図1のように8ビットの内MSBが下、LSBが上になります。つまりBASICのIMAGE/P命令と同様にすれば良いわけです。

次に、グラフィックRAMですが、これはグラフィックRAM 1、2ともそれぞれ8Kバイトの容量を持っています。この8Kバイトは、CRT上で図2のように横に40バイト、縦に200バイトとして対応しています。そしてドット・パターンは各アドレスのデータによって図のように横方向に現われます。

図1と図2のドット・パターンの違いが先ほどのA、Bの印字方法に関して問題となってきます。グラフィックRAMのデータをそのままプリンタに転送すると当然90°回転したパターンが印字されます。

この場合に注意することは、プリンタ用紙にCRTと同様のパターンを作るには、CRT画面の左下から左上の200バイト分を転送し、次に1つ右に移動して同様に上へ向って200バイト分を転送して行きます(図3)。これがBの方法です。

さて、Aの場合、つまり通常のプリンタによるキャラク

タ印字モードと同様に印字する場合には、もう1作業増えます。図4のように、グラフィックRAMのA-H番地に各々データが入っているとします。すると、CRT上のドット・パターンを同じ向き（つまりAの場合）にプリンタに転送するには、まずA-H番地のデータのビット7で作られるデータが必要で、しかもHのビット0がMSB、Aのビット0がLSBとなります。図の場合は左端の1列となりF1Hです。

以下同様ビット1、2……と繰り返します。したがって以上の作業を行なうプログラムが必要です。リストではE10FHからE167Hまでがこれに相当します。この変換を私は回転命令を使って行ないました。

さて、これでグラフィック・データについては一応説明が終わりました。ここまではBASICのCOPY/P命令と同じことです。

それでは、キャラクタの方をどうするかということになります。一言で言ってしまうとASCIIコード表にあるキャラクタのドット・パターンをプリンタ転送用の形式にして、データとして持っておけば良いわけです。

キャラクタ1個は8×8ドットで構成されているので、図4の場合と同様になります。このデータはダンプ・リストのE500HからECFFHまでに相当します（キャラクタ1個は8バイトで構成）。

なお、このデータをBASICのIMAGE/P CHR\$()+…の形式にすれば、そのままプリンタに横書きできます。

フローチャートにあるようにCRTの1行分のキャラクタを順に読み取り、そのキャラクタのドット・データをバッファに転送します。CONSOLE C80では、8バイト×80=640バイトとなります。そこで私はバッファを640バイト設けました。

バッファにキャラクタのドット・パターンが納めば、次には先ほど述べた方法でグラフィック・データを作ってバッファのデータとのORを取れば、バッファ上でグラフィックとキャラクタが合成されます。後はビット・イメージのデータとして、バッファの内容をプリンタに転送します。これを25回繰り返せばCRTの全画面のプリント・コピーができあがります。

これで基本的な説明はほとんど終わりました。

あと必要なことは次のようなことになります。

①CONSOLE C40かC80かを判断する（40字か80字か？）。

②CONSOLE NかRかを判断する（ノーマルか反転か？）。

③VRAMのアドレスを5000HからかD000Hからかのいずれかにする（5000Hの方がメモリ上有利）。

④グラフィックRAM1、2、1+2のモードのいずれの場合もコピー可能とする。

つまり、40文字、80文字のいずれでも、また反転画面はプリント・アウトも反転し、グラフィックRAM1+2とキャラクタの合成のハード・コピーが取れば、見慣れた横書きでプリンタ上に出力されて、しかも、メモリのユーザーエリアをできるだけ多く残す（このプログラムはVRAMを5000Hからにしているの、プログラムはE000Hから入っている）ということになります。

現在のところ（57年1月下旬）これらがすべて可能なプログラムはこのプログラム以外には発表されていないはずです。

自己満足はともかくとして、これらの判断をするのに必要なデータはMZ-80Bのオーナーズ・マニュアルに記載されています。ただし、VRAMのアドレスの切り換え方法ですが、初期のマニュアルには誤って記載されています。私

の所有しているものも初期のマニュアルですが、それによるとVRAMを5000Hからに割り当てるには、

IN A, (E8H)
SET 6, A
OUT (E8H, A)

となっています。

しかし、実際には、

IN A, (E8H)
SET 7, A
SET 6, A
OUT (E8H, A)

としなければ作動しません。

新版のオーナーズ・マニュアルでは後の方に訂正されています。私はこれが判るまで（秋葉原で教えてもらった）私のプログラミングのミスで動かないと思っていました。

しかし、皆さんもう安心です。このプログラムで完全なハード・コピーが取れます。

これさえできれば、MZ-80Bは高速だしFDOSもあるし、無敵です。その証拠にマイコン歴10箇月、マシン語歴2箇月、年令30才+αの私にしてこのプログラムが組めるのです。皆さん、どしどしプログラムを発表しましょう！

プログラムの説明

- ▶E023H-E02BH: VRAMを5000Hに割り当てます。
- ▶E02EH-E0B1H: メイン・ルーチン
- ▶E0B2H-E0E3H: バッファにキャラクタのドット・パターンを転送します。
- ▶E0E4H-E10EH: グラフィック・データとバッファのキャラクタ・データとのORをとるルーチン
- ▶E10FH-E167H: グラフィックRAMのデータからプリンタ転送用のデータに変換します。実際にはこのルーチン内でバッファ内のキャラクタ・データとのORを取っています。
- ▶E168H-E1E9H: バッファ内のデータをプリンタに転送します。
- ▶E1EAH-E243H: プリンタの印字、ステータス・チェックのルーチンです。この部分はMZ80-BP5のマニュアル記載のものをそのまま使用させていただきました。

サブルーチン

- ①CONSOLE C40, C80を区別する。
- ②CONSOLE N, Rを区別する。

プログラムはE000HよりECFFHまでですが、それ以外にも、バッファなどワーク・エリアがE D10H以後に設けられています。

以上のようにこのプログラムではモニタのサブルーチンをまったく使っていません。したがって、他のモニタでも使用可能なはずです。なお、FDOSで使う方は実行アドレスをE000Hに変更してください。具体的に言うと、

①CONVERT命令により（OBJ→OBJ）FDOS管理のオブジェクト・ファイルとします。

②DEBUG [CR] によりデバッグを作動させます。

③デバッグのYコマンドでこのファイルをロードします。

④Sコマンドでセーブするとき、

E000 ECFF [CR] とします。

プログラム・ダンプ・リスト (注: \$E270~\$E5EF, \$ED00~\$EFFF, \$F0C0~\$F2EFはNOP(のみ)です)

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E000 +00 32 05 EF 3E 02 03 F4 C3 20 00 00 00 00 00 32 171
E010 05 EF 3E 00 03 F4 C3 20 00 00 00 00 00 32 11E
E020 05 EF 00 00 00 00 00 00 00 00 00 00 00 00 00 00
E030 21 00 22 80 EF 21 00 66 22 00 EF 21 00 22 12C
E040 00 EF 2A 80 EF 11 10 10 32 00 2A 00 EF 11 80 17F
E050 EF 05 15 3A 05 EF 0E 20 1A 3E 0E F4 B9 2A 1F9
E060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
E070 D1 C1 0D E4 0E 4C E2 C0 00 1A 2A 0F EF 4A 42E
E080 E2 20 0A 11 2B 00 C1 BC 0E 11 50 00 19 22 80 EF 1A8
E090 2A 00 EF 11 2B 00 C1 BC 0E 11 50 00 19 22 80 EF 1A8
E0A0 EA E1 1E 1B 0D EA E1 02 0D CA E1 7A 00 EF 5D 1A8
E0B0 10 C9 05 C5 C0 42 02 05 0A 2B 0C 01 0E 0A 50 1A8
E0C0 C5 05 05 00 00 00 00 00 00 00 00 00 00 00 00
E0D0 04 19 05 20 06 F1 01 08 0B 00 00 00 25 C1 00 50 19F
E0E0 BF C1 F1 C9 05 C5 0E 28 C5 E5 11 A0 EF 0A 00 00 1A2
E0F0 00 7E 12 05 11 2B 00 19 21 05 00 2A 00 EF 1A8

```

Sum EF 49 A4 7A E6 AB 4B 39 D1 E3 F9 41 05 10 2C 9C 109

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E100 C0 0F E1 22 20 EF E1 23 C1 00 20 BC C1 F1 C9 05 1BC
E110 C5 06 00 C5 05 0A 08 0E 00 21 90 EF 00 47 20 101
E120 02 C8 C1 C0 09 22 05 25 F1 79 86 77 21 C0 4A 15B
E130 E2 2B 04 7B 86 77 21 E5 21 00 EF C0 0E C1 21 A1 EF 1FA
E140 C0 0E 21 A2 EF C0 0E 21 A3 EF C0 0E 21 A4 EF C0 16F
E150 0C 21 05 0C 0E 42 0E 3E 07 EF C0 0E 42 0E 3E 07 EF
E160 E1 C1 05 20 7E C1 F1 C9 0D 42 E2 20 04 F5 C5 0E 1EA
E170 0C 21 05 0C 0E 42 0E 3E 07 EF C0 0E 42 0E 3E 07 EF
E180 C0 E1 7E 1B C0 EA E1 7E 1B C0 EA E1 7E 1B C0 11F
E190 E1 7E 1B C0 EA E1 7E 1B C0 EA E1 7E 1B C0 EA E1 7E 1B C0 11F
E200 23 0B 7B 20 7A 00 00 00 00 00 00 00 00 00 00 00
E210 C5 7E 12 0D EA E1 10 80 02 21 10 ED 7C 0A E1 170
E220 23 0B 7B 20 7A 00 00 00 00 00 00 00 00 00 00 00
E230 C5 7E 12 0D EA E1 10 80 02 21 10 ED 7C 0A E1 170
E240 E1 7E 1B C0 EA E1 7E 1B C0 EA E1 7E 1B C0 EA E1 7E 1B C0 11F
E250 23 0B 7B 20 7A 00 00 00 00 00 00 00 00 00 00 00
E260 F1 D3 3F 7E 80 D3 7E 01 0D ED 2A 0F EF C9 189

```

Sum 97 85 AB 7B 2B 1A 0B C3 7A 27 24 7E 7E 2E C0 100

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E270 C5 05 07 1C 0C 91 10 10 20 00 00 00 00 00 00 00
E280 C1 C9 0B 0A 1F 10 10 20 00 00 00 00 00 00 00 00
E290 1E 1B C0 EA E1 7B 0C EA E1 7E 0D C2 08 1E 1C7
E2A0 0F C9 05 C5 0E 28 C5 E5 11 A0 EF 0A 00 00 1A2
E2B0 E2 09 C9 C0 DB EB C0 EF C9 00 00 05 C5 EF 5B 110
E2C0 05 0B 0F 20 10 10 ED 01 80 02 7E FF C5 E6 173 1A8
E2D0 7B 8B 01 2A 7A 00 00 00 00 00 00 00 00 00 00 00

```

Sum 90 E4 4C 07 82 50 79 3A 41 67 15 17 75 41 5B FA 100

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E2F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
E300 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
E310 00 00 07 00 00 00 00 00 00 00 00 00 00 00 00
E320 00 00 24 2A 7F 2A 12 00 00 4A 2A 10 0A 6A 82 00 153
E330 00 00 2A 7F 2A 12 00 00 4A 2A 10 0A 6A 82 00 153
E340 00 00 00 1C 22 41 00 00 00 00 00 00 00 00 00 00
E350 00 00 2A 1C 7F 1C 2A 00 00 00 00 00 00 00 00 00 00
E360 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
E370 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
E380 00 7E 51 49 49 45 3E 00 00 4A 42 7F 0A 40 00 129
E390 00 62 51 51 49 49 45 3E 00 00 4A 42 7F 0A 40 00 129
E3A0 00 10 1A 12 7F 00 00 00 27 45 45 29 11 00 00 10
E3B0 00 3C 4A 49 49 49 45 3E 00 00 03 71 09 00 00 00 117
E3C0 00 3A 49 49 49 45 3E 00 00 00 4A 49 49 49 45 3E 00 108
E3D0 00 00 00 00 24 00 00 00 00 00 80 6A 00 00 00 00 00
E3E0 00 0B 1C 3A 63 41 41 00 00 1A 14 14 14 14 14 187

```

Sum FF 5F 0A 00 F2 8D CA FF 55 92 9B 32 03 9A 9A 15B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E400 00 00 41 A3 1C 0B 00 00 02 01 81 09 00 00 00 1A8
E410 00 1C 22 49 55 59 4E 00 00 7C 0A 09 0A 0A 7C 00 1A1
E420 00 00 41 7F 49 49 49 45 3E 00 00 1C 22 41 41 22 00 0A
E430 00 41 7F 41 32 1C 00 00 7F 49 49 49 41 41 00 15C
E440 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E450 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E460 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E470 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E480 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E490 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4A0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4B0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4C0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4D0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4E0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E4F0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7

```

Sum 00 CA 6B 97 11 3C 22 00 00 4F 26 57 1B F0 4C 0E 117

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E500 00 04 02 01 02 04 00 01 01 01 01 01 01 01 01 115
E510 00 01 02 04 00 01 02 04 00 01 01 01 01 01 01 115
E520 00 7F 2B 4A 4A 4A 28 7F 00 38 4A 4A 4A 28 00 11B
E530 00 38 4A 4A 28 7F 00 38 4A 4A 4A 28 00 11B
E540 00 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08
E550 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E560 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E570 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E580 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E590 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5A0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5B0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5C0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5D0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5E0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7
E5F0 00 7F 09 09 09 01 01 00 00 1C 22 41 49 49 4A 00 1E7

```

Sum 00 1A D9 D0 30 3C BF 00 01 D0 99 3C F9 9D 91 85 1C3

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E600 00 10 08 0B 10 10 08 00 00 02 12 22 7E 10 10 12E
E610 00 00 FF 00 00 00 00 00 00 10 20 7E 10 10 12E
E620 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E630 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E640 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E650 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E660 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E670 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E680 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E690 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6A0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6B0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6C0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6D0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6E0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E
E6F0 00 00 08 0A 7E 0A 08 00 00 00 00 00 00 10 10 12E

```

```

E600 00 1C 1C 4B 7F 4B 1C 1C 00 00 00 FF 1B 1B 1B 1E4
E610 18 1F 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 167
E620 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 167
E630 18 1F 18 1F 18 18 18 18 08 08 0F 0F 0F 0F 08 08 08
E640 18 18 18 18 18 18 08 08 00 15 16 7C 16 15 00 192
E650 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 1E4
E660 00 3C 42 42 42 42 3C 00 08 08 08 08 08 00 00 190
E670 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 190
E680 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 190
E690 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 190
E6A0 00 00 FF 00 00 00 00 00 00 00 00 00 00 00 00 137
E6B0 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 17F

```

Sum 5F 9E CE 08 E2 EF 97 71 4B AE 3B A0 6A 61 90 1CB

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E6C0 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 08 15A
E6D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
E6E0 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 08 15A
E6F0 00 08 0F 08 0F 08 0F 08 0F 08 0F 08 0F 08 08 15A
E700 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E710 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E720 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E730 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E740 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E750 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E760 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E770 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E780 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E790 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7A0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7B0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7C0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7D0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7E0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142
E7F0 00 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 142

```

Sum 05 7F 8B 1A 4A E2 2C 10 D0 FF 4C BA 35 2E BE E3 07 103

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E800 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E810 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E820 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E830 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E840 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E850 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E860 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E870 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E880 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E890 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8A0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8B0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8C0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8D0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8E0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F
E8F0 00 00 02 3F 42 4A 4A 00 00 00 0A 00 00 00 00 10F

```

Sum 00 00 7F 39 6C 2D A2 00 00 00 E1 EA 49 CB 92 00 1B1

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E8F0 00 02 04 01 02 00 00 00 00 07 05 07 00 00 00 11C
E900 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E910 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E920 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E930 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E940 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E950 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E960 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E970 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E980 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E990 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9A0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9B0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9C0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9D0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9E0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A
E9F0 FF 0E 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 10A

```

Sum F1 9A 8A 3B 2E D8 D8 F1 17 AC 7A 3C 4A CA F1 1E1

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
E9F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

```

Sum FF FF FE FE 0E FE FE FE FE 00 10 08 7B 08 0A 191

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
F000 00 08 08 08 08 08 08 08 08 08 08 08 08 08 08 11C
F010 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 11C
F020 00 00 22 00 00 00 00 00 00 00 00 00 00 00 00 11C
F030 22 01 F1 2A 60 F1 22 60 F1 10 ED 2A 0A 01 11 1FD
F040 18 01 19 22 60 F1 22 60 F1 22 60 F1 22 60 F1 22 60 13B
F050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 13B
F060 19 05 20 FC E5 2A 0A F1 1E 06 08 7E 0E 00 00 100
F070 08 47 28 02 C9 C8 47 28 02 C9 C8 47 28 02 C9 C8 144
F080 C8 C9 C8 C8 C8 C8 C8 C8 C8 C8 C8 C8 C8 C8 C8 130
F090 28 02 C9 C8 47 28 02 C9 C8 47 28 02 C9 C8 47 28 02 C9 C8 144
F0A0 1A 81 12 23 E8 11 28 00 19 EB E1 10 8A 00 00 17B
F0B0 C1 F1 C8 00 00 00 00 00 00 00 00 00 00 00 00 17B

```

Sum CA 50 D6 A1 B2 4A 9D 67 B2 4F 02 1A 5B FB AB 4B 1A4

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F $I5um
F2F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F300 00 08 08 08 08 08 08 08 08 08 08 08 08 08 08 128
F310 21 00 22 80 EF 21 00 66 22 00 EF 21 00 22 12C
F320 08 1E 28 1C 0A 3C 08 00 00 8A 8A 08 10 2A 0A 102
F330 3A 48 0A 04 1A 3A 0A 04 08 10 00 00 00 00 00 100
F340 0A 08 10 10 08 0A 04 08 10 00 00 00 00 00 00 100
F350 2A 08 1C 1E 1A 3A 0A 08 00 00 00 00 00 00 00 138
F360 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 138
F370 00 00 00 00 00 00 1B 00 00 00 02 0A 10 20 00 1A6
F380 3C 42 4A 6A 62 42 3C 00 08 1B 2B 08 08 0E 0E 19C
F390 3C 42 4A 6A 62 42 3C 00 08 1B 2B 08 08 0E 0E 19C
F3A0 0A 0C 1A 2A 7A 0A 0A 07 7A 0A 7B 0A 02 4A 00 18A
F3B0 1C 20 40 7C 42 42 3C 00 08 1B 2B 08 08 0E 0E 19C
F3C0 7C 42 42 3C 00 08 1B 2B 08 08 0E 0E 19C
F3D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 138
F3E0 CE 1
```

[illegible][illegible][illegible]

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F rSum
F&F0 10 10 10 F0 10 10 10 10 10 10 10 1F 10 10 10 10 rEF
F700 F0 F0 F0 F0 F0 F7 F7 FF 00 00 00 00 70 50 70 00 rEA

```

[illegible][illegible]

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
8F0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	180
8F1	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	190
8F2	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	280
8F3	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	390
8F4	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	480
8F5	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	590
8F6	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F	680
8F7	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F	790
8F8	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	880
8F9	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F	990
8FA	100	101	102	103	104	105	106	107	108	109	10A	10B	10C	10D	10E	10F	1080
8FB	110	111	112	113	114	115	116	117	118	119	11A	11B	11C	11D	11E	11F	1190
8FC	120	121	122	123	124	125	126	127	128	129	12A	12B	12C	12D	12E	12F	1280
8FD	130	131	132	133	134	135	136	137	138	139	13A	13B	13C	13D	13E	13F	1390
8FE	140	141	142	143	144	145	146	147	148	149	14A	14B	14C	14D	14E	14F	1480
8FF	150	151	152	153	154	155	156	157	158	159	15A	15B	15C	15D	15E	15F	1590
900	160	161	162	163	164	165	166	167	168	169	16A	16B	16C	16D	16E	16F	1680
901	170	171	172	173	174	175	176	177	178	179	17A	17B	17C	17D	17E	17F	1790
902	180	181	182	183	184	185	186	187	188	189	18A	18B	18C	18D	18E	18F	1880
903	190	191	192	193	194	195	196	197	198	199	19A	19B	19C	19D	19E	19F	1990
904	200	201	202	203	204	205	206	207	208	209	20A	20B	20C	20D	20E	20F	2080
905	210	211	212	213	214	215	216	217	218	219	21A	21B	21C	21D	21E	21F	2190
906	220	221	222	223	224	225	226	227	228	229	22A	22B	22C	22D	22E	22F	2280
907	230	231	232	233	234	235	236	237	238	239	23A	23B	23C	23D	23E	23F	2390
908	240	241	242	243	244	245	246	247	248	249	24A	24B	24C	24D	24E	24F	2480
909	250	251	252	253	254	255	256	257	258	259	25A	25B	25C	25D	25E	25F	2590
90A	260	261	262	263	264	265	266	267	268	269	26A	26B	26C	26D	26E	26F	2680
90B	270	271	272	273	274	275	276	277	278	279	27A	27B	27C				

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
9F0	C1	F7	F7	F7	F7	F7	F7	FF	00	00	01	3E	54	14	14	00	145
Sum	C1	F7	F7	F7	F7	F7	F7	FF	00	00	01	3E	54	14	14	00	145

追加 キャラクターグラフィック転送プログラム

以前に発表したプログラムをマシン語に書き換えたものです。前プログラムはメイン・ルーチンをBASICで組んでいたために実行速度に難点がありました。今回のものはマシン語なので、当然のことながら格段に実行速度がアップしています。

《プログラムの機能》

このソフトは、MZ-80BのCRT表示がCONSOLE C40 (1行40文字モード)の場合に、そのキャラクタ画面の表示内容をグラフィック画面に書き込むという作業を行います。ただし、グラフィックRAM 1にだけ書き込みます。また、CONSOLE C80のときは、何もせずにリターンするの、CRT表示を乱すことはありません。

このソフトの利用法としては次のようなものが考えられます。

- ①FDOSのSA/EGR命令を行なう前にこのプログラムを実行することによって、キャラクタもまとめてディスクに保存できる。
- ②このプログラムを実行した後に、CONSOLE C80でキャラクタをC R Tに書き込めば(通常のPRINT女で)、40、80文字モードの混合したC R T表示が得られる。
- ③BASICのCOPY/P2命令でグラフィックRAM 1のLPTコピーを取ればキャラクタ+グラフィックのハード・コピーが得られる。

もともと、このプログラムはCRTのハード・コピー用に作り始めたのですが、別にハード・コピープログラムを組んだので、主として①と②の利用法が有効でしょう。

②の40、80文字モード混合のCRT画面を先に述べた。

ード・コピープログラムで処理することにより、そのまま
 の形でプリンタにハード・コピーできます。そのような利
 用法を考えて、このプログラムはF000Hから組んであります。
 ード・コピープログラムはE000Hからなので、ロード
 時に重複することなく両方のプログラムをメモリに移せ
 ます。

《プログラムのセーブ方法》

ダンブ・リストの16進データをモニタから入力してください。BASICをメインに使う人は、ジャンプ・アドレスを\$1300としてください。FDOSを使うときは、実行アドレスは\$F000にセットしてください。

《コード方法》

モニタのL命令でロードしてください。ロードの前か後にLIMIT SEFFF **CR** としてください。ハード・コピープログラムを同時にロードする場合は、LIMIT SDEFF **CR** なので間違いないようにしてください。

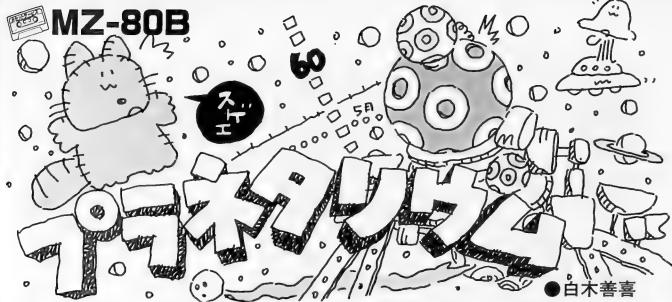
私としては両プログラムをまとめてセーブし直して使うことをお勧めします。

《使用方法》

BASICのUSR(\$F000)でこのプログラムをコールします。

《その他》

ダンプ・リストでお判りのようにF000HからF9FFH までのプログラムです。この内F200HからF9FFH までがキャラクターのドット・パターンです。



★★.....
流星のデモと北斗七星、四季の星座について、星座キーを押すと画面に星座が現れます。部屋を暗くしてみるときれいです。

★★★★.....

プログラム

プログラムはオールBASICで、約750ステップ、22.6Kバイトを使います。また、グラフィックRAM1を必要とします。

プログラムの内容

★プロローグ (流星のデモ)

星座の幕があくと、ドットで表示された星が画面にプロット (乱数で表示) され、いくつかの星がまたたき始め、しばらくすると次から次へと流星が発生し、他の星にぶつくと爆発します。流星の爆発は2～3回発生するように設定してあります。

★星座

流星のデモが終わると、図1のようにどの星座を見るか聞いてくるので、**[0]**～**[8]**のキーを押してください。

北斗七星の場合は円形の空が、他は半円の空の幕があります。画面の下部に星座や主な星の名前が表示されるので、キーを押すとその星座がプロットされます。

また、**[C]**は星座の消去、**[N]**は次の星座、**[9]**はすべて終わりを表わします。

①北斗七星

北極星と各月の北斗七星が現われます。**[1]**は北極星、**[2]**のキーを押すと何月の北斗七星と聞いてくるので、1～12までの数字を入力して**[CR]**を押してください。

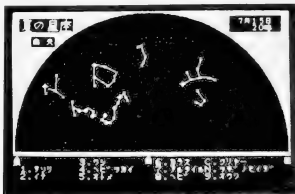
②四季の星座

春、夏、秋、冬の星座がそれぞれ北天、南天に分かれて、計8つの画面から構成されています。

なお、星座の位置の設定については、西北出版の“新小学理科のまとめ”を参考にさせていただきました。

図1

*** セ イ ラ ***	
ツキツノ イスツレカノ キーラ オシテクツツイ	
0...	北斗七星
1...	春の星座—北天
2...	春の星座—南天
3...	夏の星座—北天
4...	夏の星座—南天
5...	秋の星座—北天
6...	秋の星座—南天
7...	冬の星座—北天
8...	冬の星座—南天
9...	オワリ



```

20 REM *** 247 ***
20 REM B2.2.14 by Y.Shiraki
30 DIM X(200),Y(200),K(25),XD(25,20),YD(25,20),
35 REM *** 247 ***
47 PRINTCHR$(1)GRAPH1,C,D,J,F
48 CONSOLE$=,24
49 GOSUB10000
50 RESET 160,100
70 FORK=1TO100
80 BLINE 160,100-K,160-K,100,160,100+K,160+K,100,160,100-K
90 NEXTK
110 FORK=1TO100
120 BLINE 160+K,100-K,160-K,199,NEXTK
130 BLINE 160+K,0,160+K,199,NEXTK
135 FORV=1TO40000/NEXTV
137 PRINTCHR$(6)
140 FORJ=1TO100
150 X(J)=INT(RND(3854)*315+1)
160 Y(J)=INT(RND(3854)*195+1)
170 FORJJ=1TO600-385/NEXTJJ
180 SET X(J),Y(J)NEXTJ
190 FORV=1TO10000/NEXTV
195 GOSUB10500
199 REM *** 247 ***
200 FORK=1TO100
210 L=INT(RND(3854)*89+1)
220 N=INT(RND(3854)*89+1)
230 IF N<GOTO250
240 GOSUB10001GOTO280
250 RESET X(L),Y(L)
260 FORJJ=1TO30/NEXTJJ
270 SETX(L),Y(L)
280 NEXTK
290 L=INT(RND(3854)*89+1)
300 N=INT(RND(3854)*89+1)
310 IF N<GOTO330
320 GOSUB10500
330 GOTO350
340 SETP=
350 IF P#=""GOTO500
360 IF P#=""GOTO600
370 IF P#=""GOTO700
380 IF P#=""GOTO800
390 IF P#=""GOTO900
400 IF P#=""GOTO1000
410 IF P#=""GOTO1100
420 IF P#=""GOTO1200
430 IF P#=""GOTO1300
440 IF P#=""GOTO1400
450 GOTO340
499 REM *** 247 ***
500 GRAPHIC,FPRINTCHR$(6):SG=0
510 R=79:Y=160:Y=82:TE=1:GOSUB13100:GOSUB13200
520 FORJ=1TO632:BLINEJ,J,72,J,NEXTJ
530 CONSOLE$=,24:GOSUB2000
535 SG=9
540 GOTO330
600 GOSUB13500
610 Y=BIHV=2
615 RESTORE20040:X=BI:GOSUB13000
620 GOSUB15000
640 Y=32:HV=1
645 GOSUB15100
675 Y=BIHV=2
680 RESTORE20130:X=BI:GOSUB13000
685 RESTORE20000:X=BI:GOSUB13000
690 RESTORE20120:X=BI:GOSUB13000
695 GOSUB13000
700 GOSUB13500
710 Y=BIHV=2
715 RESTORE20040:X=BI:GOSUB13000
720 GOSUB15000
740 Y=32:HV=1
745 GOSUB15200
775 Y=BIHV=2
780 RESTORE20120:X=BI:GOSUB13000
785 RESTORE20100:X=BI:GOSUB13000
790 RESTORE20130:X=BI:GOSUB13000
795 GOSUB15000
800 GOSUB15500
810 Y=BIHV=2
815 RESTORE20050:X=BI:GOSUB13000
820 GOSUB15000
840 Y=32:HV=1
845 GOSUB15100
875 Y=BIHV=2
880 RESTORE20130:X=BI:GOSUB13000
885 RESTORE20000:X=BI:GOSUB13000
890 RESTORE20120:X=BI:GOSUB13000
895 GOSUB15000
900 GOSUB15500
910 Y=BIHV=2
915 RESTORE20050:X=BI:GOSUB13000
920 GOSUB15000
940 Y=32:HV=1
945 GOSUB15100
975 Y=BIHV=2
980 RESTORE20120:X=BI:GOSUB13000
985 RESTORE20000:X=BI:GOSUB13000
990 RESTORE20130:X=BI:GOSUB13000
995 GOSUB15000
999 GRAPHIC
1000 GOSUB13500
1010 Y=BIHV=2
1015 RESTORE20040:X=BI:GOSUB13000
1020 GOSUB15000
1040 Y=32:HV=1
1045 GOSUB15100
1075 Y=BIHV=2
1080 RESTORE20130:X=BI:GOSUB13000
1085 RESTORE20000:X=BI:GOSUB13000
1090 RESTORE20120:X=BI:GOSUB13000
1095 GOSUB15000
1100 RESTORE20130:X=BI:GOSUB13000
1105 GOSUB15000
1110 Y=BIHV=2
1115 RESTORE20040:X=BI:GOSUB13000
1120 GOSUB15000
1140 Y=32:HV=1
1145 GOSUB15200
1175 Y=BIHV=2
1180 RESTORE20120:X=BI:GOSUB13000
1185 RESTORE20100:X=BI:GOSUB13000
1190 RESTORE20130:X=BI:GOSUB13000
1195 GOSUB15000
1200 RESTORE20130:X=BI:GOSUB13000
1205 GOSUB15000
1210 Y=BIHV=2
1215 RESTORE20070:X=BI:GOSUB13000
1220 GOSUB15000
1240 Y=32:HV=1
1245 GOSUB15100
1275 Y=BIHV=2
1280 RESTORE20130:X=BI:GOSUB13000
1285 RESTORE20000:X=BI:GOSUB13000
1290 RESTORE20120:X=BI:GOSUB13000
1295 GOSUB15000
1300 GOSUB15500
1310 Y=BIHV=2
1315 RESTORE20070:X=BI:GOSUB13000
1320 GOSUB15000
1340 Y=32:HV=1
1345 GOSUB15200
1375 Y=BIHV=2
1380 RESTORE20120:X=BI:GOSUB13000
1385 RESTORE20100:X=BI:GOSUB13000
1390 RESTORE20130:X=BI:GOSUB13000
1395 GOSUB15000
1400 GOSUB15500
1499 REM *** 247 ***
1500 CONSOLE$,24:PRINTCHR$(6):CURSOR=,1:GRAPH1,C,D,J,F
1510 PRINT"*** 247 ***"
1520 CURSOR=,3
1530 PRINT"*** 247 ***"
1540 CURSOR=,5
1550 PRINT"0..."
1555 HV=2:Y=36
1560 RESTORE20000:X=72:GOSUB13000
1565 RESTORE20010:X=BB:GOSUB13000
1570 RESTORE20020:X=104:GOSUB13000
1575 RESTORE20030:X=120:GOSUB13000
1580 CURSOR=,7:Y=52
1585 PRINT"1..."I TAB(17)I"
1590 RESTORE20040:X=72:GOSUB13000
1600 GOSUB16000:GOSUB16100
1620 CURSOR=,9:Y=68
1625 PRINT"2..."I TAB(17)I"
1630 RESTORE20040:X=72:GOSUB13000
1640 GOSUB16000:GOSUB16200
1660 CURSOR=,11:Y=84
1665 PRINT"3..."I TAB(17)I"
1670 RESTORE20050:X=72:GOSUB13000
1680 GOSUB16000:GOSUB16100
1700 CURSOR=,13:Y=100
1705 PRINT"4..."I TAB(17)I"
1710 RESTORE20050:X=72:GOSUB13000
1720 GOSUB16000:GOSUB16200
1740 CURSOR=,15:Y=116
1745 PRINT"5..."I TAB(17)I"
1750 RESTORE20060:X=72:GOSUB13000
1760 GOSUB16000:GOSUB16100
1780 CURSOR=,17:Y=132
1785 PRINT"6..."I TAB(17)I"
1790 RESTORE20060:X=72:GOSUB13000
1800 GOSUB16000:GOSUB16200
1820 CURSOR=,19:Y=148
1825 PRINT"7..."I TAB(17)I"
1830 RESTORE20070:X=72:GOSUB13000
1840 GOSUB16000:GOSUB16100
1860 CURSOR=,21:Y=164
1865 PRINT"8..."I TAB(17)I"
1870 RESTORE20070:X=72:GOSUB13000
1880 GOSUB16000:GOSUB16200
1900 CURSOR=,23
1910 PRINT"9..."I"
1920 RETURN

```

1999 REM *** 8710 12741 771 ***	4070 IF P#="1" THENNN=12501:GOTO4200
2000 HV=21:Y0=1	4080 IF P#="2" THENNN=12601:GOTO4300
2005 X0=B1:RESTORE20001:GOSUB13000	4090 IF P#="3" THENNN=12701:GOTO4200
2010 X0=241:RESTORE200101:GOSUB13000	4100 IF P#="4" THENNN=12801:GOTO4200
2015 X0=401:RESTORE200201:GOSUB13000	4110 IF P#="5" THENNN=12901:GOTO4200
2020 X0=561:RESTORE200301:GOSUB13000	4120 IF P#="6" THENNN=13001:GOTO4200
2022 HV=1	4130 IF P#="7" THENNN=13101:GOTO4200
2025 X0=1571:Y0=41:RESTORE201001:GOSUB13000	4140 IF P#="8" THENNN=13201:GOTO4200
2030 X0=1571:Y0=1491:RESTORE200001:GOSUB13000	4150 IF P#="9" THENNGOSUB14100
2035 X0=2281:Y0=761:RESTORE201201:GOSUB13000	4160 IF P#="C" THENGOSUB13600
2040 X0=B41:Y0=761:RESTORE201301:GOSUB13000	4170 IF P#="N" GOTO330
2045 PRINTCHR\$(61):CURSOR3,21	4180 GOTO4050
2050 PRINT"1...87777777 / 12 "	4200 RESTORE262501:GOSUB140001:GOTO4050
2055 CURSOR3,22	4210 CURSOR1,221:PRINT"0.77777777 3.77777777 6.77777777 C.77777777"
2060 PRINT"2...87777777 / 12 "	4212 CURSOR1,221:PRINT"0.77777777 3.77777777 6.77777777 C.77777777"
2065 CURSOR3,23	4214 CURSOR1,221:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"
2067 PRINT"C...77777777 / 12 "	4216 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"
2068 CURSOR21,21	4218 CURSOR0,22
2069 PRINT"N...77777777 / 12 "	4300 GETP#
2070 CURSOR10,23	4310 IF P#="0" THENPRINTCHR\$(61):GOTO4005
2075 PRINT"12741 771 87777777 / 12 "	4320 IF P#="1" THENNN=13301:GOTO4450
2077 CURSOR21,22	4330 IF P#="2" THENNN=13401:GOTO4450
2078 PRINT"9...77777777 / 12 "	4340 IF P#="3" THENNN=13501:GOTO4450
2080 GETP#	4350 IF P#="4" THENNN=13601:GOTO4450
2090 IF P#="1" THENPRINTCHR\$(61):GOTO2130	4360 IF P#="5" THENNN=13701:GOTO4450
2095 IF P#="2" GOTO2150	4370 IF P#="6" THENNN=13801:GOTO4450
2097 IF P#="C" THENNN=631:X0=1601:Y0=B21:TE=11:GOSUB13100	4380 IF P#="7" THENNN=13901:GOTO4450
2098 IF P#="C" THENNGOSUB14100	4390 IF P#="8" THENNN=14001:GOTO4450
2100 IF P#="N" GOTO325	4410 IF P#="9" THENNGOSUB14100
2105 GOTO2080	4420 IF P#="C" THENGOSUB13600
2130 POSITION156,78	4430 IF P#="N" GOTO330
2140 PATTERN-B,A1:GOTO2045	4440 GOTO4300
2150 PRINTCHR\$(61):CURSOR3,21	4450 RESTORE263001:GOSUB140001:GOTO4300
2160 PRINT"12741 771 87777777 / 12 "	4500 FORJ=1:GOTO3001:NEXTJ
2165 CURSOR3,22	4505 CURSOR1,221:PRINT" 3.77777777 6.77777777 C.77777777"
2170 INPUT NN	4510 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"
2175 IF NN=1:GOTO2150	4520 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"
2180 TH=NN/100	4530 CURSOR0,22
2190 RESTORE250001:X0=1601:Y0=B21:GOSUB13300	4550 GETP#
2192 GOTO2045	4560 IF P#="1" THENNN=14101:GOTO4700
3000 FORJ=1:GOTO3001:NEXTJ	4570 IF P#="2" THENNN=14201:GOTO4700
3005 CURSOR1,221:PRINT"0.77777777 3.77777777 6.77777777 C.77777777"	4580 IF P#="3" THENNN=14301:GOTO4700
3100 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"	4590 IF P#="4" THENNN=14401:GOTO4700
3020 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"	4600 IF P#="5" THENNN=14501:GOTO4700
3030 CURSOR0,22	4610 IF P#="6" THENNN=14601:GOTO4700
3050 GETP#	4620 IF P#="7" THENNN=14701:GOTO4700
3060 IF P#="1" THENNN=14101:GOTO4700	4630 IF P#="8" THENNN=14801:GOTO4700
3070 IF P#="2" THENNN=14201:GOTO4700	4640 IF P#="9" THENNGOSUB14100
3080 IF P#="3" THENNN=14301:GOTO4700	4650 IF P#="C" THENGOSUB13600
3090 IF P#="4" THENNN=14401:GOTO4700	4660 IF P#="N" GOTO330
3100 IF P#="5" THENNN=14501:GOTO4700	4670 GOTO4550
3110 IF P#="6" THENNN=14601:GOTO4700	4700 RESTORE264101:GOSUB140001:GOTO4550
3120 IF P#="7" THENNN=14701:GOTO4700	5000 CURSOR1,221:PRINT" 3.77777777 6.77777777 C.77777777"
3130 IF P#="8" THENNN=14801:GOTO4700	5010 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"
3140 IF P#="9" THENNN=14901:GOTO4700	5020 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"
3145 IF P#="C" THENGOSUB13600	5030 CURSOR0,22
3147 IF P#="C" THENGOSUB13600	5050 GETP#
3150 IF P#="N" GOTO330	5060 IF P#="1" THENNN=14901:GOTO5200
3160 GOTO3050	5070 IF P#="2" THENNN=15001:GOTO5200
3200 RESTORE260101:GOSUB14000	5080 IF P#="3" THENNN=15101:GOTO5200
3210 GOTO3050	5090 IF P#="4" THENNN=15201:GOTO5200
3250 PRINTCHR\$(61)	5100 IF P#="5" THENNN=15301:GOTO5200
3260 CURSOR1,221:PRINT"0.77777777 3.77777777 6.77777777 C.77777777"	5110 IF P#="6" THENNN=15401:GOTO5200
3270 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"	5120 IF P#="7" THENNN=15501:GOTO5200
3280 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"	5130 IF P#="8" THENNN=15601:GOTO5200
3285 CURSOR0,22	5140 IF P#="9" THENNGOSUB14100
3290 GETP#	5150 IF P#="C" THENGOSUB13600
3295 IF P#="0" THENPRINTCHR\$(61):GOTO3005	5160 IF P#="N" GOTO330
3300 IF P#="1" THENNN=10901:GOTO3310	5170 GOTO5050
3310 IF P#="2" THENNN=11001:GOTO3310	5200 RESTORE264901:GOSUB14000
3320 IF P#="3" THENNN=11101:GOTO3310	5210 GOTO5050
3330 IF P#="4" THENNN=11201:GOTO3310	5500 CURSOR1,221:PRINT" 3.77777777 6.77777777 C.77777777"
3340 IF P#="5" THENNN=11301:GOTO3310	5510 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"
3350 IF P#="6" THENNN=11401:GOTO3310	5520 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"
3360 IF P#="7" THENNN=11501:GOTO3310	5530 CURSOR0,22
3370 IF P#="8" THENNN=11601:GOTO3310	5550 GETP#
3380 IF P#="C" THENGOSUB13600	5560 IF P#="1" THENNN=15701:GOTO5700
3390 IF P#="N" GOTO330	5570 IF P#="2" THENNN=15801:GOTO5700
3395 IF P#="9" THENNGOSUB14100	5580 IF P#="3" THENNN=15901:GOTO5700
3400 GOTO3290	5590 IF P#="4" THENNN=16001:GOTO5700
3410 RESTORE260901:GOSUB140001:GOTO3290	5600 IF P#="5" THENNN=16101:GOTO5700
3500 FORJ=1:GOTO3001:NEXTJ	5610 IF P#="6" THENNN=16201:GOTO5700
3505 CURSOR1,221:PRINT" 3.77777777 6.77777777 C.77777777"	5620 IF P#="7" THENNN=16301:GOTO5700
3510 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"	5630 IF P#="8" THENNN=16401:GOTO5700
3520 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"	5640 IF P#="9" THENNGOSUB14100
3530 CURSOR0,22	5650 IF P#="C" THENGOSUB13600
3550 GETP#	5660 IF P#="N" GOTO330
3570 IF P#="1" THENNN=11701:GOTO3700	5670 GOTO5550
3580 IF P#="2" THENNN=11801:GOTO3700	5700 RESTORE265701:GOSUB14000
3590 IF P#="3" THENNN=11901:GOTO3700	5710 GOTO5550
3600 IF P#="4" THENNN=12001:GOTO3700	6000 CURSOR1,221:PRINT" 3.77777777 6.77777777 C.77777777"
3610 IF P#="5" THENNN=12101:GOTO3700	6010 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"
3620 IF P#="6" THENNN=12201:GOTO3700	6020 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"
3630 IF P#="7" THENNGOSUB14100	6030 CURSOR0,22
3640 IF P#="N" GOTO330	6050 GETP#
3670 IF P#="C" THENGOSUB13600	6060 IF P#="1" THENNN=16501:GOTO6200
3680 GOTO3550	6070 IF P#="2" THENNN=16601:GOTO6200
3700 RESTORE261701:GOSUB140001:GOTO3550	6080 IF P#="3" THENNN=16701:GOTO6200
4000 FORJ=1:GOTO3001:NEXTJ	6090 IF P#="4" THENNN=16801:GOTO6200
4005 CURSOR1,221:PRINT"0.77777777 3.77777777 6.77777777 C.77777777"	6100 IF P#="5" THENNN=16901:GOTO6200
4010 CURSOR1,231:PRINT"1.77777777 4.77777777 7.77777777 9.77777777"	6110 IF P#="6" THENNN=17001:GOTO6200
4020 CURSOR1,241:PRINT"2.77777777 5.77777777 8.77777777 9.77777777"	6120 IF P#="7" THENNN=17101:GOTO6200
4030 CURSOR0,22	6130 IF P#="8" THENNN=17201:GOTO6200
4050 GETP#	6140 IF P#="9" THENNGOSUB14100
4060 IF P#="0" THENPRINTCHR\$(61):GOTO4210	6150 IF P#="C" THENGOSUB13600

ブラネタリウム プログラム・リスト

```

6160 IF P#="N" GOT0330
6170 GOT06050
6200 RESTORE246501:GOSUB14000
6210 GOT06050
6500 CURSOR1,221 PRINT " 3.747" A"ラシ 6.51747 0.7777"
6510 CURSOR1,231 PRINT " 1.777 4.777 6.777 7.777 8.777 9.777"
6520 CURSOR1,241 PRINT "2.777 3.777 4.777 5.777 6.777 7.777 8.777 9.777"
6530 CURSOR0,22
6550 GETP#
6560 IF P#="1" THENNN=17301:GOT06700
6570 IF P#="2" THENNN=17401:GOT06700
6580 IF P#="3" THENNN=17501:GOT06700
6590 IF P#="4" THENNN=17601:GOT06700
6600 IF P#="5" THENNN=17701:GOT06700
6610 IF P#="6" THENNN=17801:GOT06700
6620 IF P#="7" THENNN=17901:GOT06700
6630 IF P#="8" THENNN=18001:GOT06700
6640 IF P#="9" THENGOSUB14100
6650 IF P#="C" THENGOSUB13600
6660 IF P#="N" GOT0330
6670 GOT06550
6700 RESTORE247301:GOSUB14000
6710 GOT06550
9999 REM ### カマエイ ア"ラシ ###
10000 CURSORS,5
10010 PRINT "##### "
10015 CURSORS,6
10020 PRINT " " " #####"
10025 CURSORS,7
10030 PRINT "##### "
10035 CURSORS,8
10040 PRINT " " " #####"
10045 CURSORS,9
10050 PRINT "##### "
10055 CURSORS,10
10060 PRINT " " " #####"
10065 CURSORS,11
10070 PRINT "##### "
10075 CURSORS,12
10080 PRINT " " " #####"
10085 CURSORS,13
10090 PRINT "##### "
10095 CURSORS,14
10100 PRINT " " " #####"
10105 CURSORS,15
10110 PRINT "##### "
10120 RETURN
10500 FORJ=1TO100
10510 L=INT(RND(3854)*99+1)
10520 FORH=1TO5
10530 RESETX(L),Y(L)
10540 FORJ=1TO301:NEXTJ
10550 SETX(L),Y(L)
10560 NEXTH:NEXTL:RETURN
11000 F=INT(RND(3854)*10)
11010 IF F.5 THENX=X+1:GOT011030
1020 XX=-1
1030 FORL=1TO320:RESETX(L),Y(L)
1040 IF 0<X(L)+X:GOT011110
1050 IF 319<X(L)+X:GOT011110
1060 IF 199<Y(L)+Y:GOT011110
1070 P=X(L)+X:Q=Y(L)+Y:P=POINT(P,Q)
1080 IF (P-Q)>1:GOSUB120001:GOT011110
1090 X(L)=X(L)+X:Y(L)=Y(L)+Y
11000 SETX(L),Y(L):NEXTL
11110 RETURN
12000 RESTORE19000
12010 READ D1,D2,D3,D4
12020 IF D1=999GOT012050
12030 P1=P+D1:D1=0:D2=P+D2:D2=0:D3=P+D3:D3=0:D4=P+D4
12035 FORJ=1TO101:NEXTJ
12040 GOSUB12500
12045 GOT012010
12050 FORX=1TO5001:NEXTX
12055 RESTORE19100
12060 READ D1,D2,D3,D4
12070 IF D1=999GOT012110
12080 P1=P+D1:D1=0:D2=P+D2:D2=0:D3=P+D3:D3=0:D4=P+D4
12085 FORJ=1TO1001:NEXTJ
12090 GOSUB12700
12100 GOT012060
12110 RETURN
12500 IF P1<0 THENP1=0
12510 IF P2<0 THENP2=0
12520 IF D1<0 THEND1=0
12530 IF D2<0 THEND2=0
12540 LINEP1,D1,P2,D2
12550 RETURN
12700 IF P1<0 THENP1=0
12710 IF P2<0 THENP2=0
12720 IF D1<0 THEND1=0
12730 IF D2<0 THEND2=0
12740 BLINEL1,D1,P2,D2
12750 RETURN
12999 REM ### カマエイ ア"ラシ ###
13000 READX
13010 FORJ=1TO100
13020 READY(J)
13030 FORL=1TO101:J=J+1
13040 READXD(J,L),YD(J,L)
13050 NEXTL:NEXTJ
13055 FORJ=1TO100
13060 FORL=1TO101:J=J+1
13070 X1=X+XD(J,L):Y1=Y+YD(J,L):SHV
13080 X2=X+XD(J,L):Y2=Y+YD(J,L):SHV
13090 LINE X1,Y1,X2,Y2
13095 NEXTH:NEXTJ
13099 REM ### カマエイ ア"ラシ ###
15100 FORTH=0:TOTIE STEP0.002
15110 P=R*SIN(THN)+Q-R*COS(THN)
15120 P1=X-Q:P2=Y-Q
15130 P2=X+Q:P2=Y-Q
15140 BLINEL1,P1,D2,D2
15150 NEXTH:RETURN
15199 REM ### カマエイ ア"ラシ ###
15200 FORJ=1TO101
15210 BLINEL1,P1,D2,D2
15220 NEXTJ:RETURN
15299 REM ### カマエイ ア"ラシ ###
15300 READ LK
15310 FORJ=1TO100
15320 READ X(J),Y(J),K(J)
15325 IF 50=NGOT013400
15330 PP=X(J)-X:QD=Y(J)-Y
15340 X(J)=PP+P*COS(THN)-QD*SIN(THN)+XQ
15350 Y(J)=PP+P*SIN(THN)+QD*COS(THN)+YQ
15360 PP=X(J)-X:QD=Y(J)-Y
15370 POSIT1:QD=1
15380 DML(J)GOT013390,17400,13410
15390 PATTERN=-8,A16:GOT013420
15400 PATTERN=-7,A26:GOT013420
15410 PATTERN=-6,A36
15420 NEXTJ
15430 FORJ=1TO100
15440 LINE X(J),Y(J),X(J+1),Y(J+1)
15450 NEXTJ:RETURN
15499 REM ### カマエイ ア"ラシ ###
15500 GRAPHIC,F,PRINTCHR$(6)+CONSLES21,24:CURSOR21,24
15510 X0=160:Y0=160:R=158:TE=0.5
15520 GOSUB13100:GOSUB13200
15530 FORJ=1TO24:BLINEL1,J,72,J:NEXTJ
15540 FORJ=30704:BLINEL1,22,J,50,J:NEXTJ
15545 FORJ=47025:BLINEL1,252,J,310,J:NEXTJ
15550 RETURN
15599 REM ### カマエイ ア"ラシ ###
15600 X0=160:Y0=160:R=158:TE=0.5
15610 GOSUB13100:RETURN
15999 REM ### カマエイ ア"ラシ ###
14000 READDD
14010 IF DD=99999THENCURSOR0,221:PRINT"カマエイ ア"ラシ:RETURN
14020 IF DD=NGOT014000
14030 GOSUB13300:RETURN
14100 CURSORS0,241:CURSOR30,241:END
15000 RESTORE20090:X=241:GOSUB13000
15010 RESTORE20030:X=104:GOSUB13000
15020 RESTORE20080:X=56:GOSUB13000
15030 RETURN
15100 RESTORE20000:X=241:GOSUB13000
15110 RESTORE20110:X=40:GOSUB13000
15120 RETURN
15200 RESTORE20100:X=241:GOSUB13000
15210 RESTORE20110:X=40:GOSUB13000
15220 RETURN
16000 RESTORE20090:X=88:GOSUB13000
16010 RESTORE20030:X=104:GOSUB13000
16020 RESTORE20080:X=120:GOSUB13000
16030 RETURN
16100 RESTORE20000:X=152:GOSUB13000
16110 RESTORE20110:X=168:GOSUB13000
16120 RETURN
16200 RESTORE20100:X=152:GOSUB13000
16210 RESTORE20110:X=168:GOSUB13000
16220 RETURN
17000 Y=8
17010 X=268:RESTORE21020:GOSUB13000
17020 X=276:RESTORE21030:GOSUB13000
17030 X=284:RESTORE21040:GOSUB13000
17040 X=292:RESTORE21050:GOSUB13000
17050 Y=16
17060 X=276:RESTORE21060:GOSUB13000
17070 X=284:RESTORE21070:GOSUB13000
17080 X=292:RESTORE21080:GOSUB13000
17090 RETURN
18999 REM ### カマエイ ア"ラシ ###
19000 DATA 0,-1,0,1,0,-2,0,2,0,-1,-1,1,1,-1,1,0,-2,0,-2,-1,2,-1,-2,1,2,1
19010 DATA -1,-2,-1,2,1,-2,1,2,-2,2,-3,3,-2,3,-4,5,-1,3,-3,5,4,-2,6,-4,-3,-2,-4,-3,2,2,3
19020 DATA 1,3,3,5,0,3,3,6,-3,0,-4,0,-3,1,-5,1,-6,-5,-7,-6,-7,1,-9,1,4,3,7,5,-6,-2,-8,-4
19030 DATA -5,7,-6,8,0,-5,0,-7,-2,-5,-3,-6,0,5,0,6,2,-5,3,-6,0,8,0,11,0,-9,0,-1,1,-5,-8,-7,-10,5,-8,7,-10
19040 DATA -3,2,-6,5,-4,2,-6,4,-4,4,-8,8,-6,7,-9,10,10,0,13,0,-6,0,-11,0,999,999,999,999
19100 DATA -6,0,-11,0,10,0,13,0,-6,7,-9,10,-4,4,-8,8,-4,2,-6,4,-3,2,-6,5,5,-8,7

```

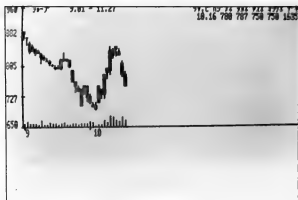
プラネタリウム プログラム・リスト

, -10, -5, -8, -7, -10, 0, -9, 0, -11
 19110 DATA 0, 8, 0, 11, 2, -5, 3, -6, 0, 5, 0, 6, -2, -5, -3, -6, 0, -5, 0, -7, -5, 7, -6, 8, -6, -2, -8,
 -4, 4, 3, 7, 5, -7, 1, -9, 1, -6, -5, -7, -6
 19120 DATA -3, 1, -5, 1, -3, 0, -4, 0, 0, 3, 6, 1, 3, 3, 5, 2, 2, 2, 3, -3, -2, -4, -3, 4, -2, 6, -4, -1,
 3, -3, 5, -2, 3, 4, -5, -2, -3, 3
 19130 DATA 1, -2, 1, 2, -1, -2, -1, 2, -2, 1, 2, 1, -2, -1, 2, -1, 0, -2, 0, 2, 1, -1, -1, 1, -1, 1, 1,
 -2, 0, 2, 0, -1, 0, 1, 0, 0, -1, 0, 1, 999, 999, 999, 999
 19999 REM *** 流石! テーブル ***
 20000 DATA 5, 1, 1, 3, 3, 3, 1, 3, 1, 3, 6, 1, 0, 7, 3, 6, 1, 5, 3, 7, 3, 3, 5, 1, 5, 7, 8, 7, 8, 6
 20010 DATA 4, 1, 2, 2, 4, 3, 1, 2, 3, 5, 4, 4, 5, 1, 1, 5, 7, 5, 1, 5, 1, 5, 8
 20020 DATA 2, 1, 1, 3, 7, 3, 2, 3, 1, 3, 5, 7, 7
 20030 DATA 9, 1, 2, 1, 2, 4, 2, 2, 1, 6, 1, 6, 4, 1, 2, 2, 5, 6, 2, 5, 1, 2, 4, 6, 4
 20032 DATA 1, 2, 4, 1, 6, 1, 1, 5, 5, 6, 5, 1, 2, 6, 6, 6, 1, 1, 7, 7, 1, 4, 4, 4, 7
 20040 DATA 9, 1, 1, 5, 1, 5, 6, 5, 1, 5, 1, 2, 2, 5, 6, 2, 5, 1, 1, 3, 5, 7, 3, 5, 2, 4, 0, 5, 4, 3, 1, 6, 1, 4, 4,
 3, 7, 6, 1, 3, 4, 5, 3, 7, 5, 2, 3, 4, 5, 4, 5, 5, 7, 5, 1, 3, 6, 5, 6, 1, 3, 5, 7, 5
 20050 DATA 9, 1, 2, 1, 7, 1, 2, 5, 1, 3, 2, 3, 5, 2, 3, 2, 6, 2, 6, 5, 1, 3, 3, 6, 1, 3, 4, 6, 4, 1, 3, 5, 6,
 5, 1, 3, 6, 2, 7, 3, 3, 6, 6, 5, 7, 3, 8, 2, 3, 6, 5, 7, 8
 20060 DATA -9, 1, 4, 1, 2, 3, 1, 2, 4, 4, 1, 3, 2, 3, 7, 1, 3, 4, 2, 7, 1, 3, 4, 4, 5, 3, 6, 1, 6, 4, 5, 6, 4,
 7, 2, 6, 4, 7, 6, 8, 7, 1, 5, 3, 5, 4, 1, 7, 3, 7, 4
 20062 DATA 7, 3
 20070 DATA 5, 1, 3, 1, 1, 3, 2, 3, 1, 5, 1, 1, 6, 2, 2, 5, 1, 5, 5, 5, 7, 5, 6, 5, 1, 3, 4, 5, 4, 5, 5, 1, 1,
 3, 5, 4, 5, 6, 5
 20080 DATA 10, 1, 4, 0, 5, 4, 2, 1, 1, 2, 7, 2, 2, 1, 2, 1, 6, 0, 8, 2, 2, 5, 2, 2, 5, 3, 1, 5, 4, 1, 2, 5, 3, 3,
 5, 4, 2, 5, 5, 2, 5, 5, 3, 4, 5, 4
 20082 DATA 1, 5, 5, 3, 6, 5, 4, 1, 4, 2, 5, 4, 7, 5, 1, 2, 5, 5, 5, 5, 5, 1, 1, 7, 5, 7, 7, 5
 20090 DATA 1, 8, 5, 2, 2, 6, 1, 4, 1, 3, 3, 2, 5, 2, 6, 3, 6, 4, 5, 6
 20100 DATA 9, 1, 2, 1, 6, 1, 4, 0, 4, 2, 1, 1, 2, 1, 7, 2, 1, 2, 7, 2, 7, 1, 2, 2, 4, 4, 1, 6, 2, 4, 4, 1, 1,
 2, 4, 6, 4, 1, 2, 6, 6, 1, 4, 4, 4, 4, 7
 20110 DATA 4, 1, 2, 1, 6, 1, 1, 1, 3, 7, 3, 2, 4, 1, 4, 4, 1, 7, 1, 4, 4, 7, 7
 20120 DATA 8, 1, 2, 1, 6, 1, 1, 1, 3, 1, 7, 2, 1, 3, 7, 3, 7, 1, 1, 5, 7, 5, 1, 1, 7, 7, 1, 4, 0, 4, 9, 1,
 4, 7, 0, 9, 1, 4, 7, 8, 9
 20130 DATA 6, 1, 1, 1, 7, 1, 1, 1, 3, 1, 7, 2, 1, 3, 7, 3, 7, 1, 1, 3, 1, 3, 7, 1, 5, 1, 5, 7, 1, 1, 7, 7, 7
 21000 DATA 2, 2, 5, 1, 2, 5, 6, 5, 1, 5, 1, 5, 7
 21010 DATA 1, 5, 3, 2, 3, 1, 6, 1, 6, 2, 5, 4, 5, 7
 21020 DATA 4, 2, 2, 1, 2, 6, 1, 7, 2, 2, 1, 6, 1, 6, 7, 1, 2, 3, 6, 3, 1, 2, 5, 6, 5
 21030 DATA 2, 2, 3, 2, 4, 1, 4, 7, 1, 3, 7, 5, 7
 21040 DATA 2, 1, 3, 1, 6, 1, 8, 3, 1, 3, 4, 4, 3, 5, 3, 6, 4, 6, 6, 5, 7, 4, 7, 3, 6
 21050 DATA 4, 1, 2, 1, 2, 7, 2, 2, 1, 6, 1, 6, 7, 1, 2, 4, 6, 4, 1, 2, 7, 6, 7
 21060 DATA 1, 9, 3, 2, 3, 5, 1, 5, 4, 1, 5, 1, 5, 5, 1, 5, 6, 2, 6, 3, 5, 5, 4, 3, 7, 6, 7
 21070 DATA 1, 8, 3, 1, 2, 2, 6, 3, 7, 5, 7, 6, 6, 6, 2, 5, 1, 3, 1
 21080 DATA 10, 1, 0, 2, 0, 6, 2, 0, 2, 2, 2, 6, 1, 0, 4, 2, 4, 1, 0, 6, 2, 6, 1, 3, 2, 7, 2, 1, 5, 1, 5, 4, 1,
 3, 4, 7, 4, 1, 3, 6, 8, 6, 1, 6, 4, 6, 8, 1, 4, 7, 4, 8
 24999 REM *** テーブル ***
 25000 DATA 7, 200, 88, 2, 208, 96, 2, 206, 108, 3, 197, 110, 3, 194, 120, 2, 188, 124, 2, 184, 136,
 2
 26010 DATA 1060, 7, 272, 64, 1, 256, 88, 2, 244, 96, 2, 232, 88, 3, 238, 78, 2, 248, 80, 3, 256, 88,
 2
 26020 DATA 1070, 5, 304, 128, 3, 296, 120, 2, 288, 116, 3, 280, 117, 3, 278, 122, 3
 26030 DATA 1080, 5, 120, 142, 2, 112, 140, 2, 116, 136, 2, 108, 136, 3, 112, 128, 3
 26040 DATA 1020, 7, 208, 72, 2, 200, 70, 2, 196, 64, 2, 188, 60, 3, 190, 54, 3, 180, 48, 2, 176, 56,
 2
 26050 DATA 1030, 7, 84, 66, 2, 81, 77, 1, 80, 85, 3, 62, 84, 3, 56, 76, 2, 76, 60, 3, 84, 66, 2
 26060 DATA 1050, 2, 160, 104, 2, 160, 104, 2
 26070 DATA 1010, 22, 208, 72, 2, 200, 70, 2, 196, 64, 2, 188, 60, 3, 190, 54, 3, 192, 40, 3, 188, 32,
 3, 168, 18, 3, 165, 21, 3, 168, 18, 3, 188, 32, 3, 192, 40, 3, 190, 54, 3
 26072 DATA 180, 48, 2, 176, 56, 2, 180, 48, 2, 152, 32, 3, 144, 24, 3, 152, 32, 3, 144, 51, 3, 150, 5,
 3, 3, 154, 48, 3
 26080 DATA 1040, 7, 160, 104, 2, 164, 105, 3, 168, 107, 3, 172, 104, 3, 177, 105, 3, 183, 100, 3, 1
 79, 97, 2
 26090 DATA 1090, 2, 208, 40, 3, 216, 45, 3
 26100 DATA 1100, 2, 81, 77, 1, 81, 77, 1
 26110 DATA 1110, 2, 272, 64, 1, 272, 64, 1
 26120 DATA 1120, 13, 174, 76, 3, 184, 80, 3, 200, 92, 3, 198, 101, 3, 191, 106, 3, 185, 110, 3, 178,
 118, 3
 26121 DATA 173, 117, 3, 170, 124, 3, 176, 128, 3, 188, 131, 3, 202, 126, 3, 201, 133, 2
 26130 DATA 1130, 9, 252, 139, 3, 248, 142, 3, 232, 136, 3, 217, 129, 3, 232, 136, 3, 230, 147, 3, 2
 24, 158, 3, 230, 147, 3, 240, 158, 3
 26140 DATA 1140, 6, 152, 122, 3, 150, 133, 3, 157, 144, 3, 165, 141, 3, 161, 129, 3, 152, 122, 3
 26150 DATA 1150, 7, 96, 120, 3, 85, 116, 2, 80, 123, 3, 85, 116, 2, 78, 114, 3, 69, 118, 3, 68, 121,
 3
 26160 DATA 1160, 2, 144, 10, 3, 146, 15, 3
 26170 DATA 1170, 3, 168, 24, 3, 168, 36, 3, 176, 48, 3
 26180 DATA 1180, 14, 152, 28, 3, 145, 28, 3, 140, 24, 3, 136, 28, 3, 137, 32, 2, 143, 38, 3, 139, 47,
 1, 143, 38, 3, 137, 32, 2, 112, 39, 3, 100, 36, 3, 112, 39, 3, 115, 38, 3, 112, 47, 3
 26190 DATA 1190, 11, 108, 64, 3, 101, 68, 3, 94, 76, 3, 88, 72, 3, 84, 64, 3, 88, 72, 3, 94, 76, 3, 89,
 88, 3, 81, 89, 3, 89, 88, 3, 88, 96, 1
 26200 DATA 1200, 12, 165, 57, 3, 161, 61, 3, 157, 56, 3, 150, 63, 3, 149, 72, 2, 144, 80, 3, 139, 80,
 3, 137, 86, 3, 130, 88, 3, 122, 108, 3, 104, 112, 3, 100, 120, 3
 26210 DATA 1210, 8, 204, 112, 2, 200, 104, 1, 190, 114, 3, 184, 118, 2, 176, 117, 2, 184, 118, 2, 1
 84, 122, 2, 192, 127, 3
 26220 DATA 1220, 8, 247, 109, 2, 252, 101, 2, 240, 88, 1, 224, 112, 2, 240, 120, 1, 247, 109, 2, 24
 1, 108, 2, 236, 107, 3
 26230 DATA 1250, 7, 115, 60, 2, 118, 70, 2, 110, 75, 2, 107, 81, 3, 100, 84, 3, 96, 96, 2, 104, 99, 2
 26260 DATA 1260, 2, 156, 120, 2, 156, 120, 2
 26270 DATA 1270, 6, 210, 50, 3, 219, 48, 3, 224, 56, 3, 214, 58, 3, 210, 50, 3, 201, 49, 1

プラネタリウム プログラム・リスト

26280 DATA 1280,7,156,120,2,160,111,3,160,104,3,157,100,3,160,93,3,152,88,3,152,96,2
 26290 DATA 1290,20,98,117,3,104,120,3,100,131,3,86,118,3,80,128,3,86,118,3,96,9
 6,2,104,99,2
 26292 DATA 96,96,2,80,91,3,74,96,3,68,108,3,74,96,3,80,91,3,96,96,2,100,84,3,10
 7,81,3,110,75,2,118,70,2,115,60,2
 26300 DATA 1300,5,190,128,2,189,135,2,184,133,2,180,136,3,175,135,3
 26310 DATA 1310,12,216,96,1,224,91,2,216,80,3,209,78,3,216,80,3,224,91,2,232,98
 ,3,235,110,3,232,98,3,224,91,2,232,86,3,248,68,3
 26320 DATA 1320,2,56,128,3,64,124,3
 26330 DATA 1330,2,92,48,3,88,56,3
 26340 DATA 1340,6,128,16,3,120,24,3,124,40,3,136,44,3,140,20,3,128,16,3
 26350 DATA 1350,2,216,96,1,216,96,1
 26360 DATA 1360,6,168,116,3,173,107,3,183,106,3,190,114,3,176,119,3,168,116,3
 26370 DATA 1370,13,136,94,3,140,80,3,152,69,3,158,67,3,161,75,3,165,82,3,169,90
 ,3,168,96,3,174,98,3,175,94,3,177,76,3,173,70,3,180,73,2
 26380 DATA 1380,2,201,49,1,201,49,1
 26390 DATA 1390,5,171,29,3,164,24,3,160,23,2,156,26,3,160,30,3
 26410 DATA 1410,11,136,104,3,134,100,3,132,95,3,127,100,1,125,106,3,121,112,2,1
 20,123,3,116,128,3,111,128,2,106,125,3,110,120,2
 26420 DATA 1420,10,98,120,3,95,121,2,96,116,3,89,112,3,87,115,2,79,115,2,72,112
 ,3,71,105,3,72,112,3,74,120,3
 26430 DATA 1430,6,60,100,3,50,93,3,52,75,3,50,93,3,38,90,1,39,96,3
 26440 DATA 1440,9,98,87,3,97,77,3,97,63,2,110,65,3,123,72,3,121,76,3,120,81,3,1
 13,87,3,97,77,3
 26450 DATA 1450,11,236,81,3,229,79,3,219,78,3,216,68,3,220,56,3,216,68,3,219,78
 ,3,208,80,3,204,72,3,208,80,3,199,85,1
 26460 DATA 1460,4,216,93,3,224,104,3,220,109,3,214,106,3
 26470 DATA 1470,2,38,88,1,38,88,1
 26480 DATA 1480,5,152,64,3,154,60,3,156,53,3,152,40,3,148,42,3
 26490 DATA 1490,5,232,32,3,225,48,2,216,43,3,225,48,2,219,64,2
 26500 DATA 1500,5,171,54,2,177,57,2,172,62,2,178,64,3,176,72,3
 26510 DATA 1510,5,208,80,2,202,85,2,208,87,3,202,85,2,192,83,3
 26520 DATA 1520,7,132,88,2,128,84,3,133,80,3,138,83,3,144,80,3,149,84,3,152,88,
 2
 26530 DATA 1530,6,229,118,2,222,108,3,200,112,1,194,118,2,204,127,3,229,118,2
 26540 DATA 1540,7,92,105,2,102,105,2,105,112,2,110,116,3,111,120,3,124,124,2,12
 8,119,2
 26550 DATA 1550,8,40,96,3,44,88,1,57,78,3,68,72,3,80,78,3,72,84,3,58,82,3,57,78
 ,3
 26560 DATA 1560,6,109,42,1,110,36,3,112,28,3,120,24,3,119,32,3,110,36,3
 26570 DATA 1570,10,136,64,3,132,67,3,112,59,3,108,52,3,112,37,2,118,34,3,112,37
 ,2,88,32,2,80,48,3,108,52,3
 26580 DATA 1580,6,195,76,3,200,69,1,213,76,3,217,88,3,213,76,3,224,68,3
 26590 DATA 1590,3,135,116,3,132,124,3,125,126,3
 26600 DATA 1600,9,44,84,3,48,88,3,43,92,3,49,95,3,62,96,2,72,98,3,85,95,3,95,97
 ,2,104,88,3
 26610 DATA 1610,6,117,90,3,120,75,3,124,72,3,128,76,3,132,72,3,145,80,3
 26620 DATA 1620,9,262,103,3,268,96,3,280,88,2,288,96,3,290,113,3,285,117,3,280,
 120,3,272,118,3,268,96,3
 26630 DATA 1630,12,244,128,2,238,127,3,239,132,3,244,136,2,250,137,3,258,131,2,
 262,129,3,269,129,1,274,138,3,270,144,3,274,138,3,279,132,3
 26640 DATA 1640,10,232,122,3,231,118,2,237,119,3,239,113,3,230,112,3,228,108,2,
 223,105,3,228,101,3,223,105,3,218,111,3
 26650 DATA 1650,8,152,16,3,155,24,3,157,30,3,152,32,3,151,39,2,144,32,2,151,39,
 2,145,48,3
 26660 DATA 1660,7,176,32,3,181,36,1,189,40,2,196,32,3,192,16,2,182,19,3,181,36,
 1
 26670 DATA 1670,22,162,136,2,165,129,2,168,127,2,173,123,3,178,126,3,192,129,3,
 200,127,3,211,116,3,209,112,3
 26672 DATA 211,116,3,200,127,3,192,129,3,178,126,3,183,119,2,176,114,2,183,119,
 2,198,96,3,204,91,3,198,96,3,192,84,3,185,92,3,190,96,3
 26680 DATA 1680,2,152,88,2,152,88,2
 26690 DATA 1690,7,152,88,2,150,90,3,149,96,3,151,101,3,148,106,3,150,112,3,155,
 108,2
 26700 DATA 1700,7,176,114,2,183,119,2,178,126,3,173,123,3,168,127,2,165,129,2,1
 62,136,2
 26710 DATA 1710,10,43,118,3,56,104,2,64,96,1,56,104,2,64,112,3,72,114,3,64,112,
 3,56,104,2,48,94,3,46,84,3
 26720 DATA 1720,5,119,65,2,121,60,2,129,63,2,132,58,2,140,60,3
 26730 DATA 1730,14,184,57,3,168,48,3,172,40,3,168,48,3,160,44,3,158,40,3,153,36
 ,3,137,22,2,153,36,3,158,40,3,160,44,3,154,46,3,150,44,1,126,33,3
 26740 DATA 1740,11,120,64,3,128,72,2,129,84,2,126,87,2,121,89,2,126,87,2,129,84
 ,2,133,97,1,119,103,2,114,72,1,128,72,2
 26750 DATA 1750,2,150,44,1,150,44,1
 26760 DATA 1760,8,104,134,3,97,137,2,95,133,2,90,136,2,95,133,2,97,128,2,99,118
 ,1,110,116,2
 26770 DATA 1770,2,76,100,1,76,100,1
 26780 DATA 1780,2,99,118,1,99,118,1
 26790 DATA 1790,2,80,92,3,76,100,1
 26800 DATA 1800,9,216,64,3,212,72,3,204,74,3,206,80,3,216,92,2,234,116,3,245,12
 0,3,254,136,2,266,140,3
 30000 DATA 99999

株式チャート・プログラム



MZ-80B



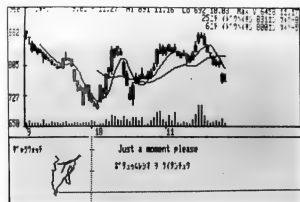
■井上信夫

私はマイコンをさわり始めて、まだ半年余りしかたっていない。株に興味がありましたので、マイコンを利用して何かプログラムが作れないかと考え、このプログラムを作成しました。

セールス・ポイント

これまでマイコン関係の雑誌に掲載されていたものに比べてチャートが大変充実しています。具体的にいえば、

- ① 表示銘柄が10銘柄まで登録可能（一般投資家にとっては充分な銘柄数である）。
- ② 表示日数が100日分、つまり株式立会日にすれば4箇月まで可能（週足で入力すれば1年52週だから2年分のデータが表示できる）。
- ③ いままでは株価の表示と出来高くらいだったが、これは移動平均線（任意の日数）がひけ、またそのカイ離率を表示できる、そのほか逆ウオッチ、ボリウム・レシオ、サイコロジカル・ラインなど多くの計算が表示できる。
- ④ チャート表示が大変美しい（フルグラフィックのため）、ただ多くの日数を表示させたいため、ローソク足をひっつけて作図するため多少見にくいとは思いますが。
- ⑤ データなどをすべてメモリに入れるためスピードが早い。
- ⑥ テープでロードさせるため非常に安価であり、一般投資家に向いている（ディスクを利用してチャートを表



示させるプログラムが市販されているが、それではハードを加えて200万円近い出費が余儀なくされる）。

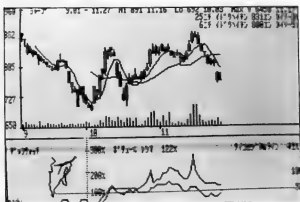
- ⑦ MZ-80Bの長所でもあるがロード時間が長い。現在6銘柄70日で約2分弱、最高でも3分以内でロードできると思います。
- ⑧ ⑥でも述べたように、G-RAM1と本体、それにテープだけでできる。つまり30万円強それに1/10の本代だけである。これで数百万円の値うちに匹敵する。これで1/10の売り上げ増（これは冗談）。

弱 点

ハード・コピーが取れない、つまり私の責任ではありませんが、MZ-80Bの弱点です。私もいろいろと聞いて回りましたが、無理であり、たとえてきてもCONSOLE40の文字しかだめだと言われています。そこを何とかあらゆる方法で良いからできればと思っています【これに関してはキャラクター・グラフィック同時コピーの項を参照してください（編）】。

LISTの説明

- 120 表示範囲の計算。
データ中の高値、安値、最高の出来高を見つけて表示させるものです。このプログラムは自分ではなかなかかわからず、シャープの「ベシッ



・「プログラミング・ガイド」を参考にしました。

- 1000 ローソク足。
陽線、陰線の違いは始値と安値をLINEすることにより自然にできるもので、わざわざ命令を作っています。また、スペース・バーを押すことにより作図を止め、途中の状態を見ることができます。再び押すことにより実行を続けます。
- 2000 作図を終えると移動平均線を何日にするのかわいてきます。表示日数以下なら何日でも指定可能ですが、終わると6日移動平均が作図されます。この2つの線の交わりから、ミニ・ゴールデンクロスorミニ・デッドクロスと言われるものがわかります。また右上に現在の移動平均値および現在値とのカイ離率が表示され、現在の株価が異常人気か否かの判別もできます。
- 3000 逆ウォッチ
時計を逆に回した形からこの名前がついたものです。つまり、朝6時の夜明けから仕の12時まで株価が上がっていくときで、昼から夕方6時そして夜と、言うふうに回転していくものです。つまり、投資家は仕ときに売って夜間に買えば良いという考え方です。
この表示は株価と出来高で作図するものです。そこで出来高、株価は日々動きが激しくなるため作図が荒くなるが多々あります。そのためそれぞれ6日間の移動平均をとり、作図しています。

6000 ボリューム・レシオ

O B V 曲線を和光証券が改良したもので、過去25日間の株価上昇日、下降日、保合い日の出来高を使用し、

$$\text{ボリューム・レシオ}(\%) = \frac{\text{株価上昇日の出来高計} + \frac{1}{2}(\text{保合い日の出来高計})}{\text{株価下降日の出来高計} + \frac{1}{2}(\text{保合い日の出来高計})} \times 100$$

の式から出るものです。

ボリューム・レシオが60～40%のゾーンを割ると、株価は底をうって反発、上昇にむかうとされています。また、突然ボリューム・レシオが一挙に上昇した場合、上昇相場のスタートとなることが多いので注意が必要です。ただ計算が面倒で二重ループを使っているので時間がかかり今後の課題です。

7000 サイコロジカル・ライン

直近12日間の中で終値の封前日比プラスの日数とマイナスの日数を数えて4勝8敗とか10勝2敗というように表現する。これを数値化して12の内プラスが何個あったかの%で数値ラインを描き%で表わしたものを、75%以上だと目先警戒、底値圏では25%以下になることが多い。

ファンクション・キーをそれぞれの銘柄のデータ・リストにしました。**[F10]**をRUNにしました。またチャートを終えると元のファンクション・キーになります。

参考文獻

池田守:「入門株式チャート」, 実業之日本社

株式チャート・プログラム・リスト

```

60 REM ***** カブカ チャート *****
61 REM Made by Nobuo Inoue Esaka Suita
62 REM November 1981
65 PRINTCHR$(6):CONSOLE, C80:GRAPH11,C,01:CLR
70 DEF KEY(1)=LIST10000-148007
71 DEF KEY(2)=LIST15000-198007
72 DEF KEY(3)=LIST20000-248007
73 DEF KEY(4)=LIST25000-298007
74 DEF KEY(5)=LIST30000-348007
75 DEF KEY(6)=LIST35000-398007
76 DEF KEY(7)=LIST40000-448007
77 DEF KEY(8)=LIST45000-498007
78 DEF KEY(9)=LIST50000-548007
79 DEF KEY(10)=RUN7
100 CURSOR0,5:PRINT""
101 PRINT"
102 PRINT"
103 PRINT"
104 PRINT"
105 PRINT"
106 PRINT"
107 PRINT"
108 PRINT"
109 PRINT"
110 PRINT"
111 PRINT"
112 CURSOR20,23:PRINT"Function key \ カブカ チャート \ data \ カブカ \ F10 \ 実行 \ CR"
115 CURSOR30,18:INPUT"ナンバ - ヲ オセ ";KO:PRINTCHR$(6)
118 DIM MD(100),OP(100),HI(100),LO(100),PR(100),VD(100),A(100),B(100),C(100),D(100),E(100),F(100),G(100),I(100)
120 REM
121 REM
122 REM
125 ON KO GOSUB140,141,142,143,144,145,146,147,148,149
126 GOTO155
140 NA$=" トウキョウ スタイル":RESTORE10000:RETURN
141 NA$=" ヲウジョウ ショウカイ":RESTORE15000:RETURN

```

```

142 NA$="タムラ セイワク" :RESTORE20000:RETURN
143 NA$="ニッホ*ン ユゼン" :RESTORE25000:RETURN
144 NA$="シターフ" :RESTORE30000:RETURN
145 NA$="N E C" :RESTORE35000:RETURN
146 NA$=" " :RESTORE40000:RETURN
147 NA$=" " :RESTORE45000:RETURN
148 NA$=" " :RESTORE50000:RETURN
149 NA$=" " :RESTORE55000
155 FOR Z=1 TO 100:READ MD(Z),OP(Z),HI(Z),LO(Z),PR(Z),VO(Z)
160 IF OP(Z)=0 THEN 180
170 NEXT Z
180 FOR X=1 TO 99:MAX=-1
190 FOR Y=XT0100
200 IF HI(Y)<=MAX THEN 220
210 MAX=HI(Y):HW=Y
220 NEXT Y
230 HI(L)=HI(X):HI(X)=MAX
240 FOR X=1 TO 99:MIN=20000
250 FOR Y=XT0100
260 IF LO(Y)>=MIN THEN 290
270 IF LO(Y)=0 THEN 310
280 MIN=LO(Y):LW=Y
290 NEXT Y
300 LO(Q)=LO(X):LO(X)=MIN
310 IF OP(X)=0 THEN 320
320 FOR X=1 TO 99:MAX=-1
330 FOR Y=XT0100
340 IF VO(Y)<=MAX THEN 360
350 MAX=VO(Y):VW=Y
360 NEXT Y
370 VO(W)=VO(X):VO(X)=MAX
380 CURSOR 7,0:PRINT NA$:MD(1);" -";MD(Z-1)
390 CURSOR 49,0:PRINT"ツキ、ヒ ハシメネ クカネ トスネ オワリネ テ"キダカ"
1000 REM
1001 REM
1002 REM
1005 DEF KEY(10)=RUN
1010 LINE 21,1,21,118,315,118
1020 FOR Y=2 TO 118 STEP 29:LINE 20,Y,23,Y:NEXT
1030 AH=INT((HI(X))/9.2)*10
1040 AL=INT((LO(LW))/10.5)*10
1050 AV=VO(X)
1060 AS=HI(X)/LO(LW)
1070 AK=AH-AL
1080 CURSOR 0,0:PRINT AH:CURSOR 0,14:PRINT AL:CURSOR 0,7:PRINT (AH+AL)/2:CURSOR 0,3:PR
INT INT((3*AH+AL)/4):CURSOR 0,11:PRINT INT((AH+3*AL)/4)
1090 DN KO GOSUB 140,141,142,143,144,145,146,147,148,149
1110 FOR Z=1 TO 100:READ MD(Z),OP(Z),HI(Z),LO(Z),PR(Z),VO(Z):IF OP(Z)=0 THEN 1300
1120 LINE 19+3*Z,118*(1-(OP(Z)-AL)/AK),21+3*Z,118*(1-(OP(Z)-AL)/AK),21+3*Z,118*(
1-(PR(Z)-AL)/AK),19+3*Z,118*(1-(PR(Z)-AL)/AK),19+3*Z,118*(1-(OP(Z)-AL)/AK)
1130 LINE 20+3*Z,118*(1-(OP(Z)-AL)/AK),20+3*Z,118*(1-(LO(Z)-AL)/AK)
1140 LINE 20+3*Z,118*(1-(PR(Z)-AL)/AK),20+3*Z,118*(1-(HI(Z)-AL)/AK)
1150 LINE 20+3*Z,118-(20*VO(Z))/AV,20+3*Z,118
1160 CURSOR 49,1:PRINT MD(Z);OP(Z);HI(Z);LO(Z);PR(Z);VO(Z);" "
1170 IF INT(MD(Z))=INT(MD(Z-1)) THEN 1190
1180 LINE 20+3*Z,118,20+3*Z,120:CURSOR(20+3*Z)/4,15:PRINT INT(MD(Z))
1190 GET SP$:IF SP$<>" " THEN 1220
1200 GET SP$
1210 IF SP$<>" " THEN 1200
1220 NEXT Z
1300 CURSOR 49,1:PRINT MD(Z-1);OP(Z-1);HI(Z-1);LO(Z-1);PR(Z-1);VO(Z-1)
1310 CURSOR 20,20:PRINT""
1320 CURSOR 0,0:PRINT AH:CURSOR 0,14:PRINT AL:CURSOR 0,7:PRINT (AH+AL)/2:CURSOR 0,3:PR
INT INT((3*AH+AL)/4):CURSOR 0,11:PRINT INT((AH+3*AL)/4)
1330 CURSOR 7,0:PRINT NA$:MD(1);" -";MD(Z-1)
1340 CURSOR 70,0:PRINT""
1350 CURSOR 32,0:PRINT"Hi";HI(HW);MD(HW);" Lo";LO(LW);MD(LW);" Max V";AV;MD(VW)
)
2000 REM
2001 REM
2002 REM
2010 CURSOR 10,23:INPUT"イト"フイキン? ";K
2015 PR(Z)=0

```

```

2020 IF K=0 THEN 2140
2030 FOR Q=0 TO K-1
2040 G(K)=PR(K-Q)/K + G(K)
2050 NEXT Q
2060 FOR Z=K+1 TO 100
2070 IF PR(Z+1)=0 THEN 2120
2080 G(Z)=(K*G(Z-1)+PR(Z)-PR(Z-K))/K
2090 G(Z+1)=(K*G(Z)+PR(Z+1)-PR(Z+1-K))/K
2100 LINE20+3#Z, 120*(1-(G(Z)-AL)/AK), 20+3*(Z+1), 120*(1-(G(Z+1)-AL)/AK)
2110 NEXT Z
2120 CURSOR 76, 1: PRINT " "
2130 CURSOR 50, 1: PRINT K;"ニチ イト"ウハイケン";INT(G(Z));"エン";" カイリ";INT(100*PR(Z)/B(Z))
-100;"Z"
2140 FORZ=6 TO 100
2150 A(Z)=(PR(Z-5)+PR(Z-4)+PR(Z-3)+PR(Z-2)+PR(Z-1)+PR(Z))/6
2160 NEXT Z
2170 FORZ=6 TO 100
2180 IF PR(Z+1)=0 THEN 2210
2190 LINE20+3#Z, 120*(1-(A(Z)-AL)/AK), 20+3*(Z+1), 120*(1-(A(Z+1)-AL)/AK)
2200 NEXT Z
2210 CURSOR 52, 2: PRINT "6ニチ イト"ウハイケン";INT(A(Z));"エン";" カイリ";INT(100*PR(Z)/A(Z))-
100;"Z"
2220 CURSOR 10, 23: PRINT " "
3000 REM
3001 REM [ キ"マワウキヤチ ]
3002 REM
3010 FORZ=6 TO 100
3020 B(Z)=(VO(Z-5)+VO(Z-4)+VO(Z-3)+VO(Z-2)+VO(Z-1)+VO(Z))/6
3030 NEXT Z
3040 LINE 88, 128, 3, 128, 3, 197, 88, 197
3050 FORZ=6 TO 100: IF PR(Z+1)=0 THEN 3080
3060 LINE 10+40*LOG(100*B(Z)/AV), 120+80*(1-(A(Z)-AL)/AK), 10+40*LOG(100*B(Z+1)/A
V), 120+80*(1-(A(Z+1)-AL)/AK)
3070 NEXT Z
3080 CURSOR 2, 17: PRINT "キ"マワウキヤチ"
6000 REM
6001 REM [ ホ"リュムレシオ ]
6002 REM
6010 LINE 95, 128, 315, 128, 315, 197, 95, 197: LINE 91, 190, 94, 190
6020 FOR Y=128 TO 197 STEP 2: SET 92, Y: NEXT Y
6030 FOR Y=140 TO 180 STEP 2: SET X, 180: NEXT Y
6040 FOR X=58 TO 310 STEP 2: SET X, 180: NEXT
6050 CURSOR 30, 17: PRINT "Just a moment please"
6060 CURSOR 29, 19: PRINT "ホ"リュムレシオ ラ タイヤンチュウ"
6070 RESTORE
6080 FOR Z=1 TO 100: READ MD(Z), OP(Z), HI(Z), LO(Z), PR(Z), VO(Z): IF PR(Z)=0 THEN 6100: NE
XT
6090 FOR X=26 TO 100: IF PR(X)=0 THEN 6180
6100 FOR S=24 TO 0 STEP -1
6110 IF PR(X-S)=<PR(X-1-S) THEN 6130
6120 E(X-1)=VO(X-S)+E(X-1): GOTO 6160
6130 IF PR(X-S)=PR(X-1-S) THEN 6150
6140 B(X-1)=VO(X-S)+B(X-1): GOTO 6160
6150 C(X-1)=VO(X-S)+C(X-1)
6160 NEXT S
6170 NEXT X
6180 CURSOR 30, 17: PRINT " "
6190 CURSOR 29, 19: PRINT " "
6200 FOR X=26 TO 100: IF PR(X+1)=0 THEN 6250
6210 D(X)=100*(E(X-1)+C(X-1)/2)/(B(X-1)+C(X-1)/2)
6220 D(X+1)=100*(E(X)+C(X)/2)/(B(X)+C(X)/2)
6230 LINE 19+3#X, 200-D(X)/5, 22+3#X, 200-D(X+1)/5
6235 CURSOR 41, 17: PRINT INT(D(X+1));"% "
6240 NEXT X
6250 CURSOR 30, 17: PRINT "ホ"リュム レシオ"
6260 CURSOR 24, 17: PRINT "300%": CURSOR 24, 20: PRINT "200%": CURSOR 24, 22: PRINT "100%"
7000 REM
7001 REM [ タイコロシ"カルライン ]
7002 REM
7010 FOR X=13 TO 100: IF PR(X)=0 THEN 7070
7020 FOR S=X-11 TO X
7030 IF PR(S)<=PR(S-1) THEN 7050
7040 F(X)=F(X)+1

```

```

7050 NEXT S
7060 NEXT X
7070 FORX=13TO100:IFPR(X+1)=0THEN7100
7080 LINE19+3*X,200-40*F(X)/12,22+3*X,200-40*F(X+1)/12
7085 CURSOR72,17:PRINTINT(100*F(X+1)/12);"%"
7090 NEXTX
7100 CURSOR60,17:PRINT"サイコロシゝカマルライン"
7110 LINE312,160,315,160:LINE312,180,315,180
7120 CURSOR74,20:PRINT"100%":CURSOR75,22:PRINT"50%"
9900 DEF KEY(1)=RUN↑
9910 DEF KEY(2)=LIST↑
9920 DEF KEY(3)=CONSOLE
9930 DEF KEY(4)=CONT↑
9940 DEF KEY(5)=AUTO↑
9950 DEF KEY(6)=CHR#(
9960 DEF KEY(7)=DEF KEY(
9970 DEF KEY(8)=GRAPH
9980 DEF KEY(9)=SAVE"
9990 DEF KEY(10)=LOAD↑
10000 REM
10001 REM      トウキョウ スマイル
10002 REM
10010 DATA9.01,972,972,960,970,25
10020 DATA9.02,960,960,955,955,11
10030 DATA9.03,975,975,965,965,41
10040 DATA9.04,975,975,975,975,25
10050 DATA9.05,970,970,970,970,22
10060 DATA9.07,970,970,970,970,22
10070 DATA9.08,980,980,980,980,12
10080 DATA9.09,970,970,970,970,2
10090 DATA9.10,970,970,950,950,6
10100 DATA9.11,950,950,950,950,6
10110 DATA9.12,940,940,929,929,26
10120 DATA9.14,928,928,925,925,7
10130 DATA9.16,920,920,920,920,9
10140 DATA9.17,921,925,910,910,31
10150 DATA9.18,920,930,915,930,6
10160 DATA9.21,920,925,920,920,8
10170 DATA9.22,920,930,920,930,13
10180 DATA9.24,920,920,920,920,3
10190 DATA9.25,925,925,925,925,48
10200 DATA9.26,926,926,925,925,14
10210 DATA9.28,925,925,910,910,30
10220 DATA9.29,909,909,909,909,11
10230 DATA9.30,900,900,890,890,12
10240 DATA10.01,891,900,880,880,14
10250 DATA10.02,890,890,890,890,20
10260 DATA10.03,900,900,900,900,3
10270 DATA10.05,900,900,900,900,3
10280 DATA10.06,906,906,906,906,2
10290 DATA10.07,906,910,906,910,6
10300 DATA10.08,950,950,950,950,1
10310 DATA10.09,999,999,999,999,38
10320 DATA10.12,1000,1020,999,1000,62
10330 DATA10.13,1000,1080,1000,1040,15
10340 DATA10.14,1060,1160,1040,1140,294
10350 DATA10.15,1130,1130,1050,1080,137
10360 DATA10.16,1020,1050,1000,1030,36
10370 DATA10.19,1050,1060,1030,1050,25
10380 DATA10.20,1030,1050,1020,1050,39
10390 DATA10.21,1040,1040,1020,1040,13
10400 DATA10.22,1020,1020,981,1020,40
10410 DATA10.23,1010,1010,1000,1000,8
10420 DATA10.24,1000,1000,1000,1000,8
10430 DATA10.26,1000,1000,981,982,26
10440 DATA10.27,990,1000,982,1000,32
10450 DATA10.28,990,997,970,970,57
10460 DATA10.29,1000,1000,1000,1000,17
10470 DATA10.30,1040,1040,1040,1040,8
10480 DATA10.31,1020,1020,1020,1020,15
10490 DATA11.02,1010,1020,1000,1000,8
10500 DATA11.04,1090,1150,1090,1150,223
10510 DATA11.05,1110,1110,1060,1100,36

```

株式チャート・プログラム・リスト

10520 DATA11.06,1090,1090,1030,1060,26	10620 DATA11.18,1230,1250,1230,1230,57
10530 DATA11.07,1120,1120,1100,1100,39	10630 DATA11.19,1220,1220,1180,1180,9
10540 DATA11.09,1120,1120,1110,1110,44	10640 DATA11.20,1230,1240,1220,1220,49
10550 DATA11.10,1140,1150,1130,1150,211	10650 DATA11.24,1250,1260,1250,1260,63
10560 DATA11.11,1150,1180,1150,1160,302	10660 DATA11.25,1250,1260,1250,1250,25
10570 DATA11.12,1170,1200,1160,1190,190	10670 DATA11.26,1250,1250,1230,1230,147
10580 DATA11.13,1200,1230,1190,1220,362	10680 DATA11.27,1230,1230,1230,1230,92
10590 DATA11.14,1230,1280,1230,1280,242	10690 DATA11.28,1230,1230,1230,1230,10
10600 DATA11.16,1260,1320,1250,1260,226	14900 DATA0,0,0,0,0,0
10610 DATA11.17,1260,1280,1250,1270,123	

株の銘柄名を変える場合は、行番号140～149のN A \$に変えたい銘柄名を入れてください。また、DATAは月日、そのときの株価の始値、高値、安値、終値、出来高の順に入れください。たとえば、12月8日のある株価の始値が972

円、高値が972円、安値が960円、終値が970円、出来高が25,000株ならば、DATA12.08,972,972,960,970,25というふうになります。DATAの区切りは行番号14900のように入ってください。

汎用データ処理プログラム

宇部鉄工マイコン・クラブ 縄田

最近プログラムレス言語があちこちで発売されていますが、我々で作って見ようと思いついたのが去年の夏ころでした。目標としてはPIPSに匹敵するものということで、Amp, Alps, PIPSの仕様を勉強して一応機能的にはまああのものデディスク・ベースで完成し、実用に供しているわけです。今回そのミニ版としてテープ・ベース版を作ったので発表します。機械語は一切使っていないので自由に改造して使ってください。

特 長

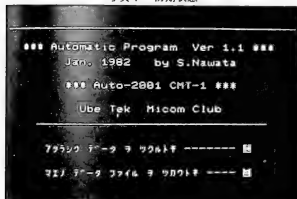
通常のものとは両面の制約から74桁程度しか処理できませんが、実用面からはプリンタの能力一歩いの136桁まで処理できるようにしました。そのため、独立したファイルを2つ用意して、あたかも1つのファイルのように自由に呼び出し、データ加工、プリントができるように工夫しています。

もう一つはAUTO-2001の名に恥じないように自動機能でデータ入力・計算・ソート・配列替え・印刷が自動的に行なえるようにした点で一度手順を登録すれば全部自分でやってくれます。

プログラム構成

次表の機能を持った2つのプログラムで構成されています。できれば1本にしたかったのですが、メモリ制約上デ

写真1 初期状態



ータ・エリアが実用的でなくなるので2本に分けました。

CMT-1はファイル作成用とサーチ、グラフ、インサツと一応の機能をそなえているので、通常の作表業務には対応できます。このCMT-1で作ったデータ・ファイルをフルに活用するの、CMT-3で、自動機能と大ファイル機能(136桁までのコントロール)を持ち、通常の事務処理はすべて自動化できます。

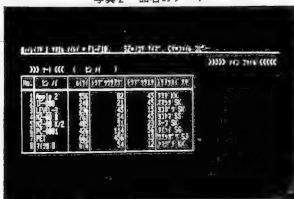
命 令	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	SZ	CL	CH	CV
機 能	表示	印刷・ ゲイン・ サブ切替	印刷・ 編集・ 配列替	ソート・ サーチ	グラフ	計算	自動	データ ファイル	マイ レイマ マナ	ノ コ リ サ イ ズ	ク リ ア	英 字 切 替	フ ァ イ ル コ ビ ー	
CMT-1	○	○	印刷	編集	サーチ	○	—	—	○	○	○	○	○	—
CMT-3	○	○	コ ビ ー	配 列 替	ソ ー ト	—	○	○	○	○	○	—	—	○

CMT-1の使用方法

最初に、**F10** をキーインすればいつでも復帰できることを憶えておいてください(エラーでBASICに抜けたときも)。

CMT-1は、データ作成と印刷・グラフ・サーチ機能を備えたもので、英女ワード・プロセッサ、通常の作表はこの

写真2 品名のソート



プログラムだけでOKです。

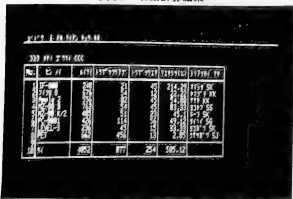
CMT-1をLOAD:RUNすると、プログラム・テープを巻き戻して「テープ、イニシャライズハ、F10ヲオシテグサイ」と表示するので、まずデータ・テープをセットしてイニシャライズします。以後はこのテープを使用し、テープ操作はすべてTOSに任せてください(けっして[F], [REW]を押さないでください)。

TOSの方式はテープを28ブロックに分けた絶対番地方式で、ジャンプは内部時計のみにより制御しています。そのため、テープが古かったり、悪かったりして回転が重くなっているとコントロールが不確実になる恐れがあるので、良品テープを使用してください。現在、シャープの標準C-15に合わせてあります。他のテープを使用する場合は6200~6236をカセットに合わせて調整してください(C-30より長いテープですと、テープの伸びなどから、コントロールが不確実になることもあります)。

また、ファイルのデリートのみを行なう、ファイルの書き込みを行なわないでジョブが終わるときは、必ずコマンドの「CL」でジョブ終わりの処理をしてください。そうしないと、TOS管理用のファイル・リストが更新されません。

- ① つぎに、新規データ・ファイル作成か前のデータ・ファイルを使うのかを聞いてきます。ここでは新規作成の「N」としましょう。
- ② 日付・ファイル名・コメントをキーインすると、これがこのファイルの個々のデータになります。以後のテープ(I/O)はこのファイル名で行なわれます。
- ③ 項目設定モードではファイルの横方向の項目を設定しますが、74桁、26項目まで設定できます。残りサイズ表示に注意しながら項目名と桁数を設定してください。なお、文書を作成する場合は「*」をキーインすると文書モードになります。設定の終りは「】」をキーインします。そこで設定OKかを聞いてきます。
- ④ 項目設定を完了するには「3」をキーインするとデータNo.1の入力待ちになります。データは255行まで入れられます。
- ⑤ 入力モードでは「h」で上と同じ。「:」で左と同じデータを入れることができます。
「←」[CR]でその行の最初から「↑」[CR]でエディット・モードになり、上の行のデータの修正ができます(←↑はカーソル・キーではない)。
空欄は「」でできますが、文書モードの場合は「」や「.」を受け付けるので、「..」をキーインすると空欄になります。
入力終了は「】」をキーインします。
- ⑥ これで命令待ちになりますが、この段階で作成されたファイルはメイン・ファイルに入っています。

写真3 自動計算結果

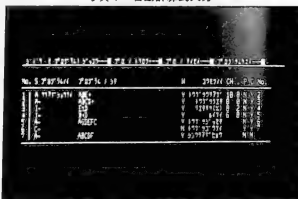


NO.	DATE	FILE	ITEM	UNIT	VALUE	UNIT	VALUE
1	1987-01-01	DATA	1	1	1.00	1	1.00
2	1987-01-01	DATA	2	1	1.00	1	1.00
3	1987-01-01	DATA	3	1	1.00	1	1.00
4	1987-01-01	DATA	4	1	1.00	1	1.00
5	1987-01-01	DATA	5	1	1.00	1	1.00
6	1987-01-01	DATA	6	1	1.00	1	1.00
7	1987-01-01	DATA	7	1	1.00	1	1.00
8	1987-01-01	DATA	8	1	1.00	1	1.00
9	1987-01-01	DATA	9	1	1.00	1	1.00
10	1987-01-01	DATA	10	1	1.00	1	1.00

サブファイルに別のデータ・ファイルを作る場合は、まず[F2]を押してサブファイルにコントロールを移します。[F2]は押すたびに切り換わり、そのときのファイル表示が右上に出ます。

- ⑦ コントロールをサブファイルにした状態でCLをキーインすると、
サブファイル・クリア…[1]、オール・クリア…[2]
と聞いてきます。メイン・ファイルのデータはそのままだにしておきます。
[1]をキーインすると新規作成か、前のファイルを使うかと聞いてくるので、[1]をキーインすればまったく同様にしてもう一つのファイルをサブファイル上に作ることができます。
- ⑧ 命令待ちにおける各命令は右上に表示しているファイルに対して行なわれるので、注意してください。
- ⑨ [F1]はファイル内容の表示です。[F1]をキーインすると、表示する行番号を聞いてくるので、
イ) 全部表示する場合は「-か-」をキーイン。
ロ) 10行までは「-10」
ハ) 20行以降は「20-」
ニ) 10行から20行は「10-20」
ホ) 15行のみは「15」
とキーインします。この行番号指定方法は他のたとえばサクジョグラフの場合も使えます。
- ⑩ 表示の途中任意のキーを押せばストップ・スタートできます。ストップさせているとき「↑」をキーインすると、エディット・モードになり、表示しているデータの修正ができます。
- ⑪ この[F1]と[F2]を交互に使えば、両ファイルの内容を表示するので、あたかも1つのファイルのように使えます(実際に1つのファイルとして自由にデータ加工できるわけですが、それはCMT-3で行ないます)。
- ⑫ 次に、[F3]をキーインしてみます。
ガメンコピー—[1] アテナインサツ—[2] データインサツ—[3] プンショインサツ—[4]
と聞いてきます。[1]をキーインすれば画面コピーを行ない、[2]をキーインすればアテナインサツになります。[3]のデータインサツと[4]の文書インサツでは、ファイル名・日付の印刷と自動ページ替えが行なわれます。
- ⑬ [F4]はデータ・ファイルを編集するためのもので、
テイセイ—[1] アペンド—[2] データ ツイカ、サクジョ—[3]
となっています。[1]をキーインすればエディット・モードになり、データだけでなく項目名も修正できます。[2]はファイルへのアペンドで、コントロール下のファ

写真4 自動計算式入力



NO.	DATE	FILE	ITEM	UNIT	VALUE	UNIT	VALUE
1	1987-01-01	DATA	1	1	1.00	1	1.00
2	1987-01-01	DATA	2	1	1.00	1	1.00
3	1987-01-01	DATA	3	1	1.00	1	1.00
4	1987-01-01	DATA	4	1	1.00	1	1.00
5	1987-01-01	DATA	5	1	1.00	1	1.00
6	1987-01-01	DATA	6	1	1.00	1	1.00
7	1987-01-01	DATA	7	1	1.00	1	1.00
8	1987-01-01	DATA	8	1	1.00	1	1.00
9	1987-01-01	DATA	9	1	1.00	1	1.00
10	1987-01-01	DATA	10	1	1.00	1	1.00

イルへ他のファイル・データをアペンドしたり、テープにある新しいデータをアペンドしたりできます。[3]はデータの追加削除で任意のデータを削除したり、任意の行番号の位置にデータを挿入したりできます。

- ①[F5]はサーチ(データ・ケンサク)でモジレツまたはスウジレツ…① ジョウケンケンサク…②と聞いてきます。[1]の場合はファイルの全データの中にある文字列または数字列を検索します。[2]をキーインするとジョウケンハ?と聞いてきます。同時に項目表を表示するので、その項目表の下に書いてあるABCで、たとえばA=100またはC>=25と次々にキーインすると、キーインされた条件のAND検索を行います。

- ③[F6]はグラフ作成で項目表を表示するとともにグラフにする項目、グラフにする範囲、グラフのフルスケールを次々にキーインすると第1項目を見出しにしたグラフを表示します。グラフ表示中任意のキーでストップ・スタートができストップ中または表示終了時に[F3]をキーインすると画面コピーします。表示終了後任意のキーを押すと、インサツするかと聞いてきて、[Y]をキーインするとプリンタに印刷します。スケールは自動スケールリングでマイナスのデータも表示します。

- ④[F9]はテープ(I/O)で、テープにセーブ① アラタニテープヨリロード②と出ます。[1]は現在コントロール下にあるファイルをテープにセーブしますが、ファイル名コメント日付を自動的にセーブできます。[2]は現在コントロール下にあるファイルを消して新しくデータをロードしますが、不用意に貴重なデータを消さないように確認しますが、なお、ロードするファイル名が分らない場合は"@"をファイル名としてキーインすると最初にあるテープ・データをロードします。

- ⑤SZをキーインするとコントロール下のファイルにデータをあと何行追加できるかを示してくれます。

- ⑥CHは英文ワード・プロセス用の英字の大小切り替えで押すたびに切り替わります。なお、ワード・プロセスはワード単位でなく行単位なので、そのように活用してください。

- 行の追加削除・検索さらに文字のエディットは自由に行なえるので、充分実用に耐えます。たとえば、文字の一括修正は検索で画面に表示してエディットで次々に修正するなど使い方は工夫してください。

- ⑦以上でCMT-1の概略の機能説明を終わります。細かい注意事項はすべて画面に表示になるので、それに従って操作してください。また、使える命令が分らない場合はメイレイ待ちモードで[CR]キーを押すとクリアされ、次にもう一度押すと使える命令表が表示され、さらに押すと再びクリアされます。

CMT-3の使用法

CMT-3はCMT-1で作成されたデータ・ファイルを元にいろいろな処理を行なうパッケージで、AUTO-2001の特長はこれで発揮できます。命令表で示してある機能のうちCMT-1と同じものは省略します。

- ①プログラムをロードすると、初めにファイル名を聞いてくるので、テープをセットしてファイル名をキーインしてください。ファイル名を忘れた場合は"@"をキ

ーイン(ここでSB-5520のバージョンが問題になります。SB-5520には2つのタイプがあるようで、1つのタイプはどんなファイル名を入れても最初のファイルをロードします。もう1つのタイプはロード機能は正常ですが圧縮印刷をするのと(136桁印刷)桁がずれます。AUTO-2001では136桁印刷を使うので、最初のタイプのSB-5520を使ってください)。

- ②[F3]は画面コピーを行ないます。

- ③[F4]は項目の配列替および追加(同時に出来ます)で現在の項目表が出るので、その項目を必要順に並べ替えることができます。また、ABC+とすると項目の追加になります。

この機能はAUTOの特長で74桁を超えた範囲まで可能で、かつコントロール下のファイルだけでなく裏ファイルの項目も自由に扱えます。我々はこれを「大ファイル・コントロール」と名付けていますが、この機能によって、見かけ上52項目の148桁までのファイル・コントロールが可能です。

つまり、表示されている表の記号を任意に選んで、たとえばAaBbfcとキーインすれば、その項目を配列替えてくれます。この場合74桁を超えると表示しませんが、印刷は136桁まで行ないます。なお、ABab+のように項目追加で印刷をする場合はInputのつどプリンタが人力待ちになります。

- ④[F5]はソート機能で、項目表に表示された項目を大きい順か小さい順に並べ替えます。この場合、ファイルは書き替えられるので注意してください。ソートは数字だけでなく文字列に対しても行なわれるので、ABC、イロハ順への並べ替えもOKです。ただし、マルチソート機能はありません。

- ⑤[F7]は計算機能で、

項目間計算① タテの合計② ヨコの合計③

となっています。[1]を押すと項目表が表示され、演算内容聞いてきます。演算結果はコントロール下のファイルに項目として追加されます。演算は大ファイルに対して行なえるので、フルにファイルを活用してください。使える演算子は"+", "-", "*", "/", "%", "へ", です。また、数字も扱えるし、負数も扱えます。たとえば、 $-0.8 * B + A$ もOKです。

小数以下何桁を出すか聞いて来るので、たとえば[2]とキーインすると小数以下3桁を4捨5入して表示します(これは計算の全ステップで行なわれるので要注意)。“/”、“%”の計算の場合分母が0のときは計算結果を0にするので、注意してください。また、“へ”を不用意に使うとすぐ桁オーバーエラーになります。

- ⑥[F8]は自動処理を行なう部分で、この機能でユーザーの工夫次第ではとどの業務がキー1つですべて自動的に行なわれます。

プログラムの実行① プログラムの登録② プログラムの修正③ プログラム・リスト表示④
がありますが、[F8]をキーインすると自動的にプログラム・リストを表示してきます。このリストは任意のキーを押すことにより、ストップ・スタートします。[1]をキーインするとスタートさせるプログラムNoを聞いてくるので、それをキーインすると、プログラムの指示に従って次々に計算・ソート・配置換え・印刷を行なってくれます。

- ⑦プログラムの登録は[2]をキーインするとプログラムに必要なデータを次々に聞いてくるので、その指示に従っ

てキーインすればOKですが、コントロールの範囲が大ファイルになっているし、書き替えを行なうとファイルの項目の位置が変わるので、あらかじめプログラムの簡単なフローチャートと項目表を作って置いて登録すれば便利で（プログラムの作り方の例は後述）。

③キーインするとEDITモードになります。不必要なプログラムを削除したり、プログラムの任意の部分を修正したりできます。

④はリスト表示で修正した後の確認に使ってください。特に、連続演算や繰り返し演算の場合、プログラムを削除すると自動的にプログラムNoがつかまるので要注意。

⑤語が元に戻りますが、F7の演算命令にタテ計とヨコ計の機能があります。ヨコの合計は特に問題ありませんが、タテ計でファイルを書き替えると実はケイの上下に区別したの横線がデータとして追加されます。これは小計と合計を1度にやるために付けたもので、テープ・ベースでは小計機能を省略しているので盲腸的存在になっています。

これが実は配列替えや項目追加のときいたずらをします。ケイの下の横線データは通常は表示・印刷のとき出てこないようにしていますが、配列替などを行なうと表示の中に出てきます。したがって、演算や自動計算処理の場合はタテ計を一番最後にやるようにプログラムをしてください。

⑥その他の機能としてはCYがあります。これはコントロールのファイルをもそのまま裏ファイルにコピーする機能で、演算などを行なってデータが変わる前を保存した演算途中のカジュアル・バッファ・ファイルが必要とするときに使います。

⑦以上で概略の説明を終わりますが、自動プログラムの登録について例をあげて説明しましょう。

ゴルフコンペのとき、Out、Inのスコアを合計してGrsを出しハンディの0.8掛けをしたものを、それから引いてNetを出し、Netの小さい順に順位を定めるといった作業は、単純作業ですが幹事任せの儀式で50人を超えるコンペではお手あげになります。一つ自動プロをさげて行って幹事に感謝せましょう。

⑧計算手順は事前に参加者の名前とハンディを入れたファイルを準備しておき、OutとInのスコアを入れればあとは $\text{Out} + \text{In} = \text{Grs}$ 、 $\text{Grs} - (\text{ハンディ}) \times 0.8 = \text{Net}$ 、Net順（小さい順）で順位判定、と単純です。

№	ナマエ	ハンディ	Out	In	Grs	Net
	(A)	(B)	(C)	(D)	(E)	(F)

という風に項目表を作っておいて作業に入ります。前準備としてはナマエとハンディのデータ・ファイルをCMT-1で作っておきましょう。

⑨まずプログラムの種類は“A”として、プログラム名は“コンペケイサン”としておきましょう。プログラムの式は“A B +”で項目追加を行ないます。書き替えは“Y”、コウモク名は“Out”桁数は“4”でよいでしょう。小数以下は関係ないので、“をキーインして空欄にします。プリントは“N”、連続計算は“Y”、連続するプログラムNoは“2”とします。これでOutのスコア入力ルーチンができました。続いて“A B C +”としてInのスコア入力を組みます。

⑩次はGrsの計算で、計算の種類は“I”、プログラムの式はC + Dとします。次のNet計算では、プログラムの式を“-0.8 * B + E”にしてください。数式式のように“E

-0.8 * B”としたところですが、このプログラムでは計算はすべて前から行なうので、注意してください。

⑪次はNetの小さい順にソートするわけですから、プログラムの種類は“S”、式は“F 1”となります。F 1の意味はFの項目を1…小さい順ということで、大きい順ならF 2になります。

⑫ついでに“天地賞”と“おしどり賞”を定めるためにOut-Inを計算しましょう。そして、ここで表示だけでなく、印刷して幹事に渡すことにしましょう。プリントシムスカは“Y”、連続計算は“N”となります。

⑬以上を打ち込んで、リストしてみてもおかしところは、エディット・モードで修正して走らせて見てください。Out Inのスコアを入れればあとは自分でやってくれるはずです。

⑭以上のプログラムを下表に示します。

№	S	プログラム名	プログラムNo	W	コウモク名	桁数	Y/N	P	C	№
1	A	コンペケイサン	AB+	Y	Out	4	N	Y	2	
2	A	~	ABC+	Y	In	4	N	Y	3	
3	I	~	C+D	Y	Grs	5	0	N	Y	4
4	I	~	-0.8*B+E	Y	Net	6	1	N	Y	5
5	S	~	F 1	Y	Net sort			N	Y	6
6	I	~	C-D	Y	テッチ、オシドリ	9	0	Y	N	

おわりに

以上で長たらしい説明を終わりますが、このプログラムの生命は大ファイルと自動機能にあるので、とにかく使ってみて皆さんでいろいろの使い方を工夫してみてください。充分市販のソフトに対抗でき、それ以上ではないかと自惚れています。

使ってみてお気付きの点やバグ（出来たてなので多少のバグとプログラムの冗長が目につくでしょう）、さらには、皆さんで改造を試みれば筆者にご連絡ください。すでにお気付きのようにCMT-2がありません。最初はディスク・ベースからのフル移植を考えて3本を考えていたのですが、編集部の要請もあって2本にしました。皆さんで是非CMT-2、予定ではフルグラフィック機能を使った多様なグラフ機能を考えていたのですが、これを作ったと思います。

また、プログラム機能の登録にあと余裕があります。ファイル・コピー、アペンド、データ削除、小計機能などのコントロール命令を工夫して、追加すればさらに機能は充実するでしょう（ただし、データ領域は小さくなります）。改造またはさらにこのプログラムを超えたものに挑戦する方で、変数表などが必要な方な編集部までご連絡ください。一緒に研究しましょう。

最後に、優秀な読者の皆さんにお願いですが、大ファイルの行きつくところInputやDisplayで同一画面上で2段で表示し、Inputもファイルを切り替えないで同一画面上でできること（もちろん、スクロールする）を考えてBASICの機能をフルに使ってみたのですが、ついにBASICだけでは成功していません。アイデアがありましたら教えてください。

AUTO-2001 CMT-1 プログラム・リスト

```

0 REM
0 REM      >>>> Auto-2001 CMT-1 <<<<<
0 REM - UBE Tek. Micom Club - by S.Nawata
0 REM
1 REM---INITIAL SET-----
2 FORI=1TO10:POKE4544+I,15:I:NEXT
3 CLR:TEMPO7:CONSOLE$0,24,C40:CURSOR$0,11:PRINT" *** シェアウ オマチ フグ ヲイ。 ***":CURSOR
9,7:PRINT">>> Auto-2001 CMT <<<"
4 DIMD$(1,255),I$(1,26),S$(1,26),IC(1,26),IP(1,26),IN(1),IE(1),DE(1),F1$(1),F2$
(1),F3$(1),F4$(1),F5$(1),F6$(1),F7$(1),T$(1,26),T(1)
5 DIML$(1),FDB(26),K$(28),X(26),Y(26),W(26):IC(1,1)=1:FORI=1TO28:K$(I)=""O":NEX
T:L$(I)="" ***
6 DEF KEY(1)=ヒキ
7 DEF KEY(2)=ワツ
8 DEF KEY(3)=イン
9 DEF KEY(4)=ノ
10 DEF KEY(5)=リキ
11 DEF KEY(6)=クツ
12 DEF KEY(7)=キキ
13 DEF KEY(8)=キキ
14 DEF KEY(9)=チーフ
15 DEF KEY(10)=GOTO997
16 REM:GOSUB6400:USR(0F14):CURSOR$,15:PRINT">>>テーク テーフ ヲ ヒョウシテ スキヤ キー ヲ オシテフク
"ヤ<<<":PRINT"PRINT" *** テーフ ノ イニシャルイズ" F10 ヲ オシテ フグ ヲイ ***"
17 GETA$:IFA$="" THEN17
18 IFASC(A$)=25THEN3060
19 GOSUB6000
20 USR(0F14):PRINTCHR$(6):PRINTSTRING$( "",39):PRINT:PRINT" *** Automatic Pro
gram Ver 1.1 ***":PRINT:PRINTTAB(8):"Jan. 1982 by S.Nawata":PRINTCHR$(1)
21 PRINTTAB(8):" *** Auto-2001 CMT-1 ***+CHR$(1)+CHR$(1):PRINTTAB(10):"Ube Tek
Micom Club"+CHR$(1):PRINTTAB(4):STRING$( "",51):CHR$(1)
22 CONSOLE$15,24:PRINT:PRINTTAB(5):"アタラク テーグ ヲ フグキキ -----" +CHR$(1)+CHR$(
1):PRINTTAB(5):"エノ テーグ ヲ フグキキ -----" +CHR$(1)+CHR$(1)
23 IFC3=1THENCURSOR$,23:PRINT( "ファイル キエトキ") F2 ヲ キーイン)":GOSUB5980:USR(0F14)
24 GETA$:IFA$="" THEN34
25 IFASC(A$)=17THENIFF=0THENF=1:GOTO32
26 IFASC(A$)=17THENIFF=1THENF=0:GOTO32
27 IFC3=1THENIFASC(A$)=25THEN99
28 A=VAL(A$):IF(A>2)+(A=0)THENGOSUB5000:GOTO34
29 IFA=2THEN82
30 PRINTCHR$(6):TAB(7):"エン カ" ヒ キ イレテ フグ ヲイ。":PRINT:PRINTTAB(14):"INPUT" エン =
":A$:Z=RIGHT$( " "+A$,2):C$=Z$+" /":CURSOR$1,0:PRINTC$:IFA$="" GOTO997 THEN32
41 CURSOR$14,19:INPUT"ツキ =" :A$:A$=RIGHT$( " "+A$,2):Z$=Z$+A$:C$=C$+A$+" /":CURSOR
$1,0:PRINTC$:CURSOR$15,21:INPUT"ヒ =" :A$
42 A=RIGHT$( " "+A$,2):Z$=Z$+A$:C$=C$+A$:CURSOR$1,0:C$=LEFT$(C$+SPACE$(30),8):P
RINTC$:CURSOR$10,23:INPUT"ヤリオシマスカ (Y/N) ? ":A$:IFA$<"N" THEN32
44 PRINTCHR$(6):CONSOLE$18,24:INPUT"ファイルメイ (6 ヲシ)" :A$:IFA$="" GOTO997 THEN32
45 FT$(F)=LEFT$(A$+SPACE$(30),6):CURSOR$,0:PRINTFT$(F):CURSOR$,20:INPUT" コメント (
15 ヲシ)" :A$:A$=LEFT$(A$,15):IFLEFT$(A$,7)=GOTO997 THEN32
46 FT$(F)=LEFT$(FT$(F)+A$+SPACE$(30),21):CURSOR$,0:PRINTLEFT$(FT$(F),6):" " :RI
GHT$(FT$(F),15):FT$(F)=FT$(F)+Z$
47 CURSOR$10,23:INPUT"ヤリオシ マスカ (Y/N) ":A$:IFA$="" GOTO997 THEN32
48 IFA$<"N" THEN44
49 REM-----ITEM SET-----
50 DE(F)=0:CONSOLE$0,24,C80:Y=12:IE(F)=0:F1$(F)="" :F2$(F)="" :F3$(F)="" :F4$(F)=""
:F5$(F)="" :L$=""
51 GOSUB5100:PRINT:PRINTTAB(22):">>> コツワツ ヒョウイ (オワリ) ヲ キーイン <<<":CURSOR$,10:
PRINTSTRING$( "",79):GOSUB5980
52 CURSOR$21,6:PRINT"----- アソビ ワケイ" (F) ヲ キーイン -----":CURSOR$22,9:PRINT" コツワ
ツ ヒョウ" (F) (F) "コツワツメイ" ト スルコト
53 CURSOR$22,11:PRINT" コツワツメイ":TAB(45):"クダラフ (74クダラフ)":CONSOLE$13,24
54 FORI=1TO26:IFL$="" THENPRINTCHR$(6):GOTO71
55 IFY=23THENCURSOR$,24:PRINT:Y=Y-1
57 CURSOR$,Y+1:PRINT" コツワツ ( " :CHR$(64+I):" ) " :INPUT" :A$:IFA$="" THEN17
58 IFA$="" THENCURSOR$,9:PRINTSPACE$(79):CURSOR$,11:PRINTSPACE$(79):CURSOR$16,3:
PRINT"
>>> アソビ ワケイ <<<+SPACE$(70):L$=A$:GOTO95
60 IFLEFT$(A$,1)="" THEN32:THENS(F,I)="" :S" :I$(F,I)=RIGHT$(A$,LEN(A$)-1):GOTO64
62 S(F,I)="" :N" :I$(F,I)=A$:IFA$="" GOTO997 THEN32
64 PRINTCHR$(2):TAB(45):" INPUT" :A$:IFA$="" GOTO997 THEN32
65 IF(ASC(A$)<48)+(ASC(A$)>57)THENGOSUB5000:GOTO64
66 D=VAL(A$)+1
67 IC(F,I)=IE(F)=IE(F)+D:Y=Y+1:IFIE(F)>74THENGOSUB5300:PRINT:PRINTTAB(9):" ***
コツワツ クダラフ ノ オハリ" テーグ ヲ ヒョウシテ フグ ヲイ ***":FORI=0TO10000:NEXTI:GOTO50
68 CURSOR$,6:PRINTSPACE$(79):GOSUB5400:CURSOR$,5:PRINT" -----" +F1$(F):CURSOR$,6:
PRINT" | No. "+F2$(F):CURSOR$,7:PRINT" | -----" +F3$(F):IN(F)=I
69 CURSOR$,6,11:PRINTSPACE$(18):CURSOR$,6,11:X=73-IE(F):IFX<0THENX=X+1
70 PRINT" X ノゾクノクダラフ =" :X: " " :NEXTI
71 USR(0F14):CURSOR$7,24:PRINT" ヒョウ" ヲ ヒョウシ スル -----" ヒョウイ ヲ ヤリオス -----" ヒョウイ
ヲ オワル -----"
72 GETA$:IFA$="" THEN72
74 A=VAL(A$):IF(A>3)+(A=0)THENGOSUB5000:GOTO72

```

AUTO-2001 CMT-1 プログラム・リスト

```

75 CURSOR0,23:PRINTSPACE$(79)
76 IFA=2THENS0
78 IFA=3THENC3=0:GOTO99
80 CURSOR5,6:GOSUB5200:GOSUB5600:CURSOR9,23:GOTO71
81 REM-----FILE LOAD-----
82 C3=1
83 GOSUB6100:GOSUB7100:INPUT"ロート" Ⅱ ファイル No. = " :A$:IFC1=1THENIFA$="GOTO99" THE
N1500
84 IFA$="GOTO99" THENC3=0:GOTO32
85 IFA$="0" THENGOSUB7100:INPUT" ファイル 名 = " :FC$:FC$=LEFT$(FC$,4) " Ⅱ Ⅱ Ⅱ :GOTO
89
86 A=VAL(A$):IF(A>LE)+(A=0) THENGOSUB5000:GOTO83
88 FC$=LEFT$(L$(A),6)+Z:A:GOSUB6200:GOSUB6700
89 F1$(F)="":F2$(F)="":F3$(F)="":F4$(F)="":F5$(F)="":F6$(F)="":ROPE/TFC$:GOSUB
7100:PRINT" "+FC$+" " フォント 番号 :FORI=1TO26:INPUT/TA$,B$,C$:IFA$="I-END" THEN91
90 I$(F,I)=A$:S$(F,I)=B$:IC(F,I)=VAL(C$):IE(F)=IE(F)+IC(F,I):GOSUB5400:IN(F)=I:
NEXTI
91 FORI=1TO255:INPUT/TD$(F,I):IFD$(F,I)="D-END" THENINPUT/TFT$(F):DE(F)=I-1:GOTO
93
92 NEXTI
93 FORI=1TO26:INPUT/TT$(F,I):IFT$(F,I)="C-END" THENCLOSE/T:REW:CONSOLE0,24,CB0:
T(F)=I-1:GOTO96
94 NEXTI
95 S$(F,1)="S":I$(F,1)=LEFT$(LEFT$(FT$(F),6)+" "+MID$(FT$(F),7,15)+SPACE$(70),
61)+MID$(FT$(F),22,2)+" / "+MID$(FT$(F),24,2)+" / "+RIGHT$(FT$(F),2):D=70:GOTO67
96 IFC1=1THEN99
97 CURSOR0,24:FORI=0TO16:PRINT:FORJ=0TO30:NEXT:NEXTI:CONSOLE0,24
98 CURSOR25,9:PRINT">>> Auto-2001 CMT-1 <<<:CURSOR24,12:PRINT"*** Ube Tek Mic
on Club ***"
99 GOSUB7000:GOSUB5500:IP(F,1)=1:F6$(F)="I" Ⅱ :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,
I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI
102 F6$(F)=F6$(F)+SPACE$(IC(F,IN(F))-1)+"I"
109 IFC3=0THENC3=1:GOTO200
111 GOSUB5980
112 C1=1:CURSOR0,0:USR(%OF14):INPUT"" :A$:A$=LEFT$(A$,2)
113 IFA$="GO" THEN97
120 FORI=1TO25STEP2:IFA$=MID$("ヒョウフンノヘンチシズク Clc1",I,2) THEN140
130 NEXTI:GOTO150
140 C2=0:ON(I+1)/2GOTO400,600,800,1000,1500,170,170,180,180,2000,2500,3000,3000
150 IFC2=1THEN99
155 CURSOR0,2:PRINT"F1=ヒョウフン F2=メインメモリ F3=インメモリ F4=ハードディスク F5=テープ F6=マイク"
160 CURSOR0,3:PRINT"F6=7" フォント 番号 :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI
170 CHANGE:GOTO99
180 W=SIZE:W=(W-560)/(IE(F)+5):W=INT(W):IFW<0THENW=0
182 GOSUB7100:CURSOR0,0:PRINT"/コリサズ" Ⅱ フォント 番号 :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI
184 GETA$:IFA$=" " THEN184
186 GOTO99
199 REM-----INPUT-----
200 A$="I":IFDE(F)>0THENGOSUB7100:CURSOR0,0:PRINT" ヒョウフンノヘンチシズク Ⅱ DL=7777" フォント 番号 :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI
205 IF(A$="DL")+(A$="dl") THEN250
210 CURSOR0,0:PRINT" ヒョウフンノヘンチシズク " :IFLEFT$(A$,2)="GO" THEN99
215 A=VAL(A$):IF(A<1)+(A>DE(F)+1) THENGOSUB5000:GOTO200
220 CONSOLE2,24:CURSOR0,3:PRINT" >>> ヒョウフンノヘンチシズク <<<:GOSUB5700:GOSUB7100:CURSOR9
,0:PRINT" フォント 番号 :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI
221 CONSOLE7,24:Y=7
222 USR(%OF14):IF(DE(F)=255)+(SIZE<500) THENGOTO5800
224 A$="I":IF(A$="I")+(A$="i") THEN250
225 FORI=1TOIN(F):CURSOR0,Y:PRINT" "+RIGHT$( " "+STR$(A),3)+"I":TAB(IP(F,I)+4)
:IFS$(F,I)="S" THENIFIC(F,I)=0THENGOSUB5200:B$=MID$(C$,6,70):GOTO234
226 IFS$(F,I)="S" THENINPUT"" :B$:GOTO234
228 INPUT"" :C$
230 IFC$="GOTO99" THEN99
232 B$=RIGHT$(SPACE$(70)+C$,IC(F,I)-1)
234 IFLEFT$(B$,7)="GOTO99" THEN99
236 IFLEFT$(B$,2)=" " THEN99
237 IF(LEFT$(B$,1)="I")+(C$="I") THENIF I=1 THEN246
238 IF(LEFT$(B$,1)="S")+(RIGHT$(B$,1)="I")+(RIGHT$(C$,1)="I")+(RIGHT$(X$,1)="I")
THENPRINTCHR$(2):SPACE$(79):CHR$(2):GOTO224
239 IF(LEFT$(B$,1)="I")+(C$="I") THENPRINTCHR$(2):SPACE$(79):CHR$(2):GOSUB5900:G
OTO224
240 IF(LEFT$(B$,1)="I")+(C$="I") THENB=0:FORJ=0TOI-1:B=B+IC(F,J):NEXTJ:B$=MID$(D
$(F,A-1),B+1,IC(F,I))
241 IF(LEFT$(B$,1)="I")+(C$="I") THENB$=MID$(A$,IP(F,I-1),IC(F,I-1)-1):C$="I"
243 A$=A$+LEFT$(B$+SPACE$(70),IC(F,I)-1)+"I":CURSOR0,Y:PRINT" "+RIGHT$( " "+STR
$(A),3)+"I":NEXTI:Y=Y+1:IFY=24THENPRINT:Y=Y-1
244 IFA=DE(F)+1THENDS(F,A)=A$:A=A+1:DE(F)=A-1:GOTO222
245 FORI=DE(F) TOASTEP-1:D$(F,I+1)=D$(F,I):NEXTI:D$(F,A)=A$:A=A+1:DE(F)=DE(F)+1:
GOTO222
246 CURSOR0,Y:GOTO5150
250 CURSOR0,0:PRINT" フォント 番号 :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6$(F)=F6$(F)+SPACE$(IC(F,I)-1)+"I":NEXTI

```

```

252 A=DE(F):GOSUB5750
254 GETA$:IFA$="" THEN254
256 IF(A$<>"Y")*(A$<>"y") THEN99
258 IFL+M>DE(F) THENDE(F)=L-1:GOTO264
260 FORI=LTODE(F):PRINTI:CHR$(2):IFI=DE(F)+1-MTHENDE(F)=DE(F)-M:GOTO264
262 D$(F,I)=D$(F,I+M):NEXTI
264 FORI=DE(F)+1TOA:PRINTI:CHR$(2):D$(F,I)="" :NEXTI:GOTO99
399 REM-----DISPLAY-----
400 CURSOR0,0:PRINT" ヒョウシ " :IFF=1THENGOSUB5980
424 GOSUB5250:IFA$=""GOTO99 THEN99
426 CONSOLE$2,24:GOSUB7100:PRINT" ニンノ キーヲ オシテ ストップ*スタート シマス。 (ストップ*ノット
*「フ」=フイ(フ),F10=/イ(イチ) " :Y=7:CONSOLE$7,24
427 CURSOR0,3:PRINT" >>> ヒョウシ <<< "
429 GOSUB5700:FORI=LTOI+M-1:IFI=DE(F)+1THEN5150
430 IF(I=DE(F))*(RIGHT$(D$(F,I),1)=""4") THEN5150
432 GETA$:IFA$="" THEN438
434 GETA$:IFA$="" THEN434
435 IFA$=""4" THENGOSUB5900:CURSOR0,Y:PRINTSPACE*(79):GOTO450
436 IFASC(A$)=25THEN99
438 GOSUB5501:NEXTI:GOTO5150
450 IFY=24THENY=Y-1:GOTO434
452 GOTO434
549 REM-----DISP SUB-----
550 IFY=24THENPRINTY=Y-1
555 CURSOR0,Y:PRINT" | "+RIGHT$( " " +STR$(I),3)+" | "+D$(F,I):Y=Y+1:RETURN
600 IFF=0THENF=1:G=0:GOTO99
610 F=0:G=1:GOTO99
799 REM-----PRINTER-----
800 GOSUB7100:CURSOR0,0:PRINT" インサツ Ⅱ **** フリプリンター / スイッチ ハ ハイチ イマスカ ?
****":PRINT/PCHR$(17)
805 GOSUB7100:PRINT" カンメン*プリント --- Ⅱ アナ インサツ --- Ⅲ テータ インサツ --- Ⅳ フォンショ インサツ
--- Ⅴ"
810 GETA$:IFA$="" THEN810
815 IFASC(A$)=25THEN99
820 A=VAL(A$):IF(A>4)+(A=0) THENGOSUB5000:GOTO810
825 IFA=3THEN878
827 IFA=4THEN910
830 IFA=1THENCURSOR0,0:PRINTSPACE*(159):CURSOR0,2:COPY/P1:CONSOLE$0,24:GOTO99
840 IFCONSOLE$2,24:CURSOR0,0:PRINT" アナインサツ Ⅱ コツエカク 「ナメ」,「フ」ヨ
ツシヨ,「フ」ヒ「ン」コ「フ」ノ シ「ン」ニ ヲ「フ」イマスカ (Y/N) ?"
850 GETA$:IFA$="" THEN850
860 IF(A$<>"Y")*(A$<>"y") THEN99
862 PRINT/PCHR$(6):GOSUB7100:PRINT" アナ インサツ ヨツシ ヲ セット シテ ト「レカ スキ キー ヲ オシテ クラ
「タイ。"
864 GETA$:IFA$="" THEN864
866 IFASC(A$)=25THEN99
868 FORI=1TODE(F):PRINT/PMID$(D$(F,I),IP(F,3),IC(F,3)-1):PRINT/PMID$(D$(F,I),IP
(F,2),IC(F,2)-1)
870 FORJ=1TOIC(F,1):IFMID$(D$(F,I),IC(F,1)-J,1)>"0" THEN874
872 NEXTJ
874 PRINT/P:PRINT/PTAB(5):MID$(D$(F,I),IP(F,1),IC(F,1)-J)+ " ヤマ":PRINT/P:PRINT/
P:NEXTI:GOTO99
878 GOSUB5250:IFA$=""GOTO99 THEN99
881 GOSUB7100:INPUT" キョウカヲ アタ イナツ ニ シマスカ (Y/N) ? ":Z$=IF(Z$=""Y")+(Z$=""y")T
HEND=40:GOTO887
882 D=80
887 K=0:GOSUB7100:PRINT" ファイルメイ ヒツケ ヲ イナツ シマスカ (Y/N) ? ":CONSOLE$2,24:GOSUB51
00
888 GETA$:IFA$="" THEN888
889 IF(A$=""Y")+(A$=""y") THENPRINT/PLEFT$(FT$(F,6)," " :MID$(FT$(F,7,15):TAB(70
):MID$(FT$(F,22,2):"/":MID$(FT$(F,24,2):"/":RIGHT$(FT$(F,2):GOTO891
890 GOTO892
891 PRINT/P:STRINGS(" ",78):PRINT/P:PRINT/P
892 A=0:A$=LEFT$(F1$(F),LEN(F1$(F))-1):B$=LEFT$(F3$(F),LEN(F3$(F))-1):C$=LEFT$(
F5$(F),LEN(F5$(F))-1)
893 PRINT/P" " +A$+" " :PRINT/P" | No.Ⅱ "+F2$(F):PRINT/P" | " +B$+"4" :FORI=LTOI
+M-1:IFI=DE(F)+1THEN900
894 IF(I=DE(F))*(RIGHT$(D$(F,I),1)=""4") THEN900
895 IF(Z$=""Y")+(Z$=""y") THENPRINT/PF6$(F)
896 PRINT/P" | "+RIGHT$( " " +STR$(I),3)+" | "+D$(F,I):A=A+1:IF(A/D)=INT(A/D)
THENGOSUB905
898 NEXTI
900 PRINT/P" | " +C$+" " :FORI=0TO6:PRINT/P:NEXTI:GOTO99
905 PRINT/P" | " +C$+" " :FORC=0TO14:PRINT/P:NEXTC:PRINT/P" | " +A$+" " :PRINT/
P" | No.Ⅱ "+F2$(F):PRINT/P" | " +B$+"4" :RETURN
910 CONSOLE$2,24:CURSOR0,0:PRINT" フォンショ Ⅱ ":GOSUB5250:IFA$=""GOTO99 THEN99
924 K=0:GOSUB7100:PRINT" Ⅱ ファイルメイ ヒツケ ヲ イナツ シマスカ (Y/N) ? ":CONSOLE$2,24:PRINT
" " :I$(F,1):PRINT" " :STRINGS(" ",70)
926 GETA$:IFA$="" THEN926
928 IF(A$=""Y")+(A$=""y") THENPRINT/P" " +I$(F,1):PRINT/P" " +STRINGS(" ",71
):PRINT/P:PRINT/P:K=1
930 A=0:FORI=LTOI+M-1:IFI=DE(F)+1THEN940
935 PRINT/P" " +LEFT$(D$(F,I),LEN(D$(F,I))-1):PRINT/P:A=A+1:IF(A/40)=INT(A/4

```

AUTO-2001 CMT-1 プログラム・リスト

```

0) THENGOSUB950
938 NEXTI
940 FORJ=0TO6:PRINT/P:NEXTJ:GOTO99
950 IFK=0THENFORC=0TO18:PRINT/P:NEXTC:RETURN
955 FORC=0TO14:PRINT/P:NEXTC:PRINT/P" "+I$(F,1):PRINT/P" "+STRING$( "",
71):PRINT/P:PRINT/P:RETURN
999 REM-----EDITING-----
1000 GOSUB7100:CURSOR0,0:PRINT" ヲシヨウ Ⅱ テイセイ --- Ⅲ アハント --- Ⅳ テーパ ツイカ、ワクシヨ
--- Ⅴ"
1010 GETA$:IFA$="" THENI010
1020 IFASC(A$)=25THEN99
1030 A=VAL(A$):IF(A>3)+(A=0) THENGOSUB5000:GOTO1010
1040 IFA=1THEN4000
1050 IFA=2THEN4100
1060 IFA=3THEN200
1500 GOSUB7100:CURSOR0,0:PRINT" テーフ Ⅱ テーフニ セーフ ---Ⅲ アラタ テーフヨリ ロート ---Ⅳ フ
イル リスト ---Ⅴ ファイル ワシ"ヨ ---Ⅵ"
1510 GETA$:IFA$="" THENI510
1520 IFASC(A$)=25THEN99
1530 A=VAL(A$):IF(A>4)+(A=0) THENGOSUB5000:GOTO1510
1549 IFA=1THEN1560
1550 IFA=2THENGOSUB5750:GOTO1800
1551 IFA=3THENGOSUB6100:GOTO1500
1552 IFA=4THENGOSUB7100:INPUT" ワシ"ヨ スル ファイル No. = " :Z$:IFZ$="GOTO99" THEN99
1553 Z=VAL(Z$):IF(Z>LE)+(Z=0) THENGOSUB5000:GOTO1552
1554 GOSUB6230:GOTO1500
1560 GOSUB7100:CURSOR0,0:PRINT" セーフ Ⅱ シタ ファイルメイ ト ヒツ"ク テーパ シマスカ (Ⅷ/Ⅸ) ?
" :CONSOLE2,24:CURSOR0,20:GOSUB5100: GOSUB5980
1570 GETA$:IFA$="" THENI570
1580 IFASC(A$)=25THEN99
1590 IF(A$="Y")+(A$="y") THENFC$=LEFT$(FT$(F),6):GOTO1690
1600 CURSOR0,24:INPUT" ファイルメイ ( 6 ヲシ" ) = " :A$:IFA$="GOTO99" THEN99
1610 FC$=LEFT$(A$+SPACE$(6),6):INPUT" コソント ( 15 ヲシ" ) = " :A$:FC$=LEFT$(FC$+
A$+SPACE$(15),21)
1620 INPUT" ワシ"ヨ ツシ ( 2 ヲシ" ) = " :A$:A$=RIGHT$( " "+A$,2):FC$=FC$+A$
1622 INPUT" ツシ ( 2 ヲシ" ) = " :A$:A$=RIGHT$( " "+A$,2):FC$=FC$+A$
1624 INPUT" ヒ ( 2 ヲシ" ) = " :A$:A$=RIGHT$( " "+A$,2):FC$=FC$+A$:F
C$=L$(LE+1)=FC$:FC$=LEFT$(FC$,6):GOTO1690
1630 WOPEN/LEFT$(FC$,6):CURSOR0,24:PRINTLEFT$(FC$,6)+" " :カキコミ ヲウツ
1640 FORI=1TOIN(F):PRINT/IT$(F,I),S$(F,I),STR$(IC(F,I)):NEXTI:PRINT/T" I-END","X
X","XX"
1650 FORI=1TODE(F):PRINT/TD$(F,I):NEXTI:PRINT/T" D-END":PRINT/TFT$(F)
1655 FORJ=1TOT(F):PRINT/TT$(F,J):NEXTJ:PRINT/T" C-END":CLOSE/T:REW:PRINT:PRINT" *
*** ヲシ"ラツ ヲツ ク"ワシ. ****":USR$(OF14)
1660 GOSUB6400:WOPEN/T" ファイル リスト":PRINT:PRINT" ファイル リスト カキコミ ヲウツ"
1665 FORI=1TOLE:PRINT/TL$(I):NEXTI:PRINT/T" ****","XX","XX":FORI=1TO28:PRINT/TK$(
I):NEXTI:CLOSE/T:REW:GOTO97
1690 Z=0:FORI=1TOLE:IFFC$=LEFT$(L$(I),6) THENZ=I
1692 NEXTI:IFZ<>0 THENGOTO1695
1693 L$(LE+1)=FT$(F):GOSUB6210:IFLE=0 THEN99
1694 GOSUB6300:GOTO1630
1695 PRINT:PRINT:PRINT:PRINT" オナシ ナマエ ノ ファイルカ ステ"ニ アリマスカ:PRINT:INPUT" カキカエ マスカ
(Y/N) ? " :A$:IFA$<"Y" THEN99
1696 GOSUB6230:GOTO1693
1800 GETB$:IFB$="" THENI800
1804 IF(B$<>"Y")*(B$<>"y") THEN99
1808 FORI=0TODE(F):D$(F,I)="" :NEXTI:IE(F)=0:F1$(F)="" :F2$(F)="" :F3$(F)="" :F4$(F
)=" :F5$(F)="" :F6$(F)="" :DE(F)=0
1809 GOTO83
2000 GOSUB7100:CURSOR0,0:PRINT" ワーチ Ⅱ ヲシ" マリハ スウシ"レツ ケンザク ---Ⅲ シ"ヨウケン ケンザク
---Ⅳ" :CURSOR0,3:GOSUB5050
2042 GETA$:IFA$="" THEN2042
2044 IFASC(A$)=25THEN99
2046 A=VAL(A$):IF(A>2)+(A=0) THENGOSUB5000:GOTO2042
2048 IFA=2THEN2100
2050 GOSUB7100:INPUT" モシ"レツ マリハ スウシ"レツ ? " :B$:IFB$="GOTO99" THEN99
2051 GOSUB7100:PRINT" ファイル カキカエ マスカ (Ⅷ/Ⅸ) ?"
2052 GETA$:IFA$="" THEN2052
2053 IFASC(A$)=25THEN99
2054 IF(A$="Y")+(A$="y") THENA$="Y":A=0
2060 CONSOLE2,24:GOSUB7100:PRINT" :B$;" :カケザク ヲウツ." :CURSOR0,3:PRINT" >>> ク
ソツ <<< ( :B$;" ) :B=1:GOSUB5700:CONSOLE2,24
2062 FORI=1TODE(F):FORJ=1TOIE(F):IFMID$(D$(F,I),J,LEN(B$))=B$ THENGOSUB2090:GOTO
2066
2064 NEXTJ:GOTO2068
2066 IFA$="Y" THENA=A+1:D$(F,A)=D$(F,I)
2068 PRINTI:CHR$(2):NEXTI
2069 IFA$="Y" THENDE(F)=A
2070 IFB=1 THENPRINTCHR$(6):
2071 GOTO5150
2090 IFB=1 THENPRINTCHR$(6):B=0
2092 PRINT" | "+RIGHT$( " "+STR$(I),3)+" | "+D$(F,I):RETURN
2100 CONSOLE2,9:LE=1:Z$="シ"ヨウケン --- " :B=1:GOSUB7100:PRINT" C=15 マリハ B<=200 /ヨウニ

```

```

2100 CURSOR0,4:GDSUB5050
2102 CURSOR0,2:PRINTSPACE$(79):PRINTSPACE$(77):CURSOR0,2:PRINTZ$:INPUT"ジョウケン ハ
? ":A$=IFA$="GOTO99" THEN99
2104 IFA$="J" THEN2132
2106 K=1:C$=MID$(A$,2,1):B$=MID$(A$,2,2):A=ASC(LEFT$(A$,1))-64
2108 IFB$="<" THENJ=4:GOTO2122
2110 IF(B$="=")+(B$="<") THENJ=5:GOTO2122
2112 IF(B$=">")+(B$="=") THENJ=6:GOTO2122
2114 IFC$="=" THENJ=1:GOTO2124
2116 IFC$="<" THENJ=2:GOTO2124
2118 IFC$=">" THENJ=3:GOTO2124
2120 GDSUB5000:GOTO2102
2122 K=K+1
2124 IFK=1:Z$=Z$+A$+" "
2126 X$=MID$(A$+SPACE$(20),K+1,IC(F,A)-1):IFS$(F,A)="N" THENX$=RIGHT$(A$,LEN(A$)-
(K):X$=RIGHT$(SPACE$(50)+X$,IC(F,A)-1):W(L)=1:GOTO2130
2128 W(L)=0
2130 FD$(L)=X$:Y(L)=A:Y(L)=A:Y(L)=J:L=L+1:GOTO2102
2132 USR$(OF14):GDSUB7100:PRINT" ファイル ヲ カキカエ マスカ (値/置) ?"
2134 GETY$:IFY$="" THEN2134
2136 IFASC(Y$)=25 THEN99
2137 IFY$="Y" THENC=0
2138 GDSUB7100:PRINT" "Z$;" ヲ カンザク チョウ." :CONSOLE$2,24:CURSOR0,3:PRINT" >>> ク
ンザク <<< ( "Z$;" )":GDSUB5700:CONSOLE$7,24
2140 FORM=1:TODE(F)+2:PRINTM:CHR$(2)
2142 FORN=1:TOL-1:I=X(N):X$=FD$(N):J=Y(N):W=W(N):IFM=DE(F)+1 THENGOTO2195
2144 Z$=MID$(D$(F,M),IP(F,I),IC(F,I)-1)
2146 ONJGOTO2140,2158,2160,2176,2181,2186
2148 IFW=0 THEN2154
2150 IFVAL(Z$)=VAL(X$) THEN2191
2152 GOTO2194
2154 IFZ$=X$ THEN2191
2156 GOTO2194
2158 IFW=0 THEN2164
2160 IFVAL(Z$)<VAL(X$) THEN2191
2162 GOTO2194
2164 IFZ<X$ THEN2191
2166 GOTO2194
2168 IFW=0 THEN2174
2170 IFVAL(Z$)>VAL(X$) THEN2191
2172 GOTO2194
2174 IFZ>X$ THEN2191
2176 GOTO2194
2178 IFW=0 THEN2179
2177 IFVAL(Z$)<>VAL(X$) THEN2191
2178 GOTO2194
2179 IFZ<>X$ THEN2191
2180 GOTO2194
2181 IFW=0 THEN2184
2182 IFVAL(Z$)<=VAL(X$) THEN2191
2183 GOTO2194
2184 IFZ<=X$ THEN2191
2185 GOTO2194
2186 IFW=0 THEN2189
2187 IFVAL(Z$)>=VAL(X$) THEN2194
2188 GOTO2194
2189 IFZ>=X$ THEN2191
2190 GOTO2194
2191 NEXTN:IFB=1 THENPRINTCHR$(6):B=0
2192 PRINT"|" +RIGHT$( "STR$(M),3)+"|" +D$(F,M)
2193 IFY$="Y" THENC=C+1:D$(F,C)=D$(F,M)
2194 NEXTM
2195 IFY$="Y" THENDE(F)=C
2196 IFB=1 THENPRINTCHR$(6):B=0
2197 GOTO5150
2500 CURSOR0,3:GDSUB5050:GDSUB7100:CURSOR0,0:INPUT" クラフ 〓 (ライシヨノ コウエク カ ミダ
シ ナリマス) クラフ ニ スル コウエク ? ":C$:IFC$="GOTO99" THEN99
2520 K=ASC(C$)-64:IF(K<0) THENC=C+1:IF(K<0) THENGDSUB5000:GOTO2500
2530 GDSUB5250:IFA$="GOTO99" THEN99
2540 GDSUB7100:INPUT"(キョウシノ トキョウ F3=カメン フリント) フル スケール ライデ クラフタイ." :CC$:I
FCC$="GOTO99" THEN99
2550 D=LEN(CC$):N=VAL(CC$)
2560 W=K:A=INT((76-IC(F,1))/4)*4:CONSOLE$0,24:PRINTTAB(20);"*** "I$(F,K);"/
クラフ ***:PRINTTAB(20);STRING(" ",IC(F,K)+17)
2590 PRINTTAB(IC(F,1)-1);" ";SPACE$(A/2-D);N/2:PRINTTAB(IC(F,1)+A-D-1);CHR$(2);
N
2600 PRINTTAB(IC(F,1)-1);" ";STRING(" ",A/4-1);"+";STRING(" ",A/4-1);"+";STR
ING(" ",A/4-1);" ";STRING(" ",A/4-1);" ";CONSOLE$4,24
2605 FORI=L:TOL+M-1:IFI=DE(F)+1 THEN2621
2607 PRINTTAB(IC(F,1)-1);" ";SPACE$(A/4-1);" ";SPACE$(A/4-1);" ";SPACE$(A/4-1);
" ";SPACE$(A/4-1);" ";CHR$(2)
2610 IFI=L THENCURSOR0,4
2612 GETA$:IFA$="" THEN2619

```

AUTO-2001 CMT-1 プログラム・リスト

```

2614 GETA$=IFA$="" THEN2614
2615 IFASC(A$)=18THENGOSUB2810: COPY/P1:FORI=0TO4:PRINT/P:NEXT:GOTO99
2619 PRINTLEFT$(D$(F,I),IC(F,I)):B=VAL(MID$(D$(F,I),IP(F,W),IC(F,W)-1)):IFB<0
HENB=ABS(B):C=INT((B/N)*A):PRINTSTRING$( "=",C)+ (マウス):NEXTI
2620 C=INT((B/N)*A):PRINTSTRING$( "-",C):NEXTI
2621 GOSUB2810
2622 GETA$=IFA$="" THEN2622
2624 IFASC(A$)=18THENCOPY/P1:FORI=0TO4:PRINT/P:NEXT:GOTO99
2630 GOSUB7100:CURSOR25,0:INPUT " イマツ ヲツカ (Y/N) ? ":Y$=IFY$<"Y" THEN99
2700 PRINT/PCHR$(17):PRINT/PTAB(20):*** "I$(F,K):" / グラフ ***:PRINT/PTAB(2
0):STRING$( " ",IC(F,K)+17):PRINT/P:PRINT/P
2702 PRINT/PTAB(IC(F,1)-1):"0":SPACE$(A/2-D):N/2:TAB(IC(F,1)+A-D-1):CHR$(2):N
2710 PRINT/PTAB(IC(F,1)-1):"1":STRING$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":ST
RING$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":
2720 FORI=L:TOL+M-1:IFI=DE(F)+1THEN2770
2730 PRINT/PTAB(IC(F,1)-1):"1":SPACE$(A/4-1):"1":SPACE$(A/4-1):"1":SPACE$(A/4-1
):"1":SPACE$(A/4-1):"1"
2750 PRINT/PLEFT$(D$(F,I),IC(F,I)):B=VAL(MID$(D$(F,I),IP(F,W),IC(F,W)-1)):IFB<
0THENB=ABS(B):C=INT((B/N)*A):PRINT/PSTRING$( "=",C)+ (マウス):GOTO2765
2760 C=INT((B/N)*A):PRINT/PSTRING$( "-",C):
2765 PRINT/PTAB(IC(F,1)-1)+A$:"1":NEXTI
2770 PRINT/PTAB(IC(F,1)-1):"1":SPACE$(A/4-1):"1":SPACE$(A/4-1):"1":SPACE$(A/4-1
):"1":SPACE$(A/4-1):"1"
2771 PRINT/PTAB(IC(F,1)-1):"1":STRING$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":ST
RING$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":
2772 FORI=0TO4:PRINT/P:NEXTI:GOTO99
2810 PRINTTAB(IC(F,1)-1):"1":STRING$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":STRI
NG$( "-",A/4-1):"1+":STRING$( "-",A/4-1):"1+":RETURN
3000 GOSUB7100:CURSOR0,0:PRINT " クリマ " :I:IFF=0THENPRINT " メイン ":GOTO3004
3002 PRINT " クリマ " :I
3004 PRINT " ファイル クリマ ---" :I オール クリマ ---" :I ショワ クリマ ---" :I
3010 GETA$=IFA$="" THEN3010
3020 IFASC(A$)=25THEN99
3030 A=VAL(A$):IF(A>3)+(A=0) THENGOSUB5000:GOTO3010
3040 IFA=1THENCURSOR0,0:PRINTSPACE$(79):GOTO32
3050 IFA=2THENRUN
3060 CONSOLESO,24:PRINT " ***** シムラウ オマツ クマ ヲイ *****"
3062 WOPEN/T " ファイル リスト ":PRINT:PRINT " ファイル リスト クリマ ツク"
3064 FORI=1TOLE:PRINT/TL$(I):NEXTI:PRINT/T " ***** ", "XX", "XX":FORI=1TO28:PRINT/TK$(
I):NEXTI:CLOSE/T:REW:END
3070 CONSOLESO,24:END
3999 REM----UTILITY-----
4000 GOSUB7100:CURSOR0,0:PRINT " ED " :I テイセイ ) 1 キョウ コトニ オカイマス."
4010 CURSOR0,0:USR($OF14):IFIC(F,1)=70THENGOSUB5200:GOSUB5600:C$=CC$:GOSUB5910:
GOTO4030
4020 GOSUB5905:CC$=C$:GOSUB5600
4030 IFLEFT$(X$,6)="GOTO99" THENX$="" :GOTO99
4032 CURSOR0,0:GOTO4000
4100 GOSUB7100:CURSOR0,0:PRINT " アハント " :I テーフ ファイル クリマ ツク ---" :I :IFF=0THENPRI
NT " クリマ " :I:G=1:GOTO4102
4101 PRINT " メイン ":I:G=0
4102 PRINT " ファイル クリマ ツク ---" :I
4104 GETA$=IFA$="" THEN4104
4106 IFASC(A$)=25THEN99
4108 A=VAL(A$):IF(A>2)+(A=0) THENGOSUB5000:GOTO4104
4109 IFA=1THEN4200
4110 FORI=DE(F)+1TODE(F)+DE(G):IFSIZE<500THEN5800
4111 PRINTI:CHR$(2):D$(F,I)=D$(G,I-DE(F)):NEXTI:DE(F)=DE(F)+DE(G):GOTO99
4200 GOSUB6100:GOSUB7100:INPUT " アハント " :I ファイル No. = ":A$:IFA$="" GOTO99" THEN99
4205 A=VAL(A$):IF(A>LE)+(A=0) THENGOSUB5000:GOTO4200
4210 FC$=LEFT$(L$(A),6):Z=A:GOSUB6200:GOSUB6300:ROPEN/TFC$:GOSUB7100:PRINT " "
+FC$+" クリマ ツク"
4220 FORI=1TO26:INPUT/TA$,B$,C$:IFA$<"I" THENNEXT
4225 FORI=DE(F)+1TO25:INPUT/T D$(F,I):IFD$(F,I)="" D-END" THENCLOSE/T:DE(F)=I-1:R
EW:GOTO99
4230 NEXTI
4999 REM----UTILITY-----
5000 MUSIC"+BOR+BR+B" RETURN
5050 PRINT " " +F1$(F):PRINT " [No.]+F2$(F):PRINT " " +F3$(F):B$="" " :I:FORI
=1TOIN(F):B$=B$+SPACE$(IC(F,I)-1)+CHR$(64+I):NEXTI
5060 PRINTB$:PRINTSPACE$(79):RETURN
5100 PRINT " Title " :I:LEFT$(FT$(F),6):" " :MID$(FT$(F),7,15):TAB(64):"Date " :I:M
ID$(FT$(F),22,2)+" / "+MID$(FT$(F),24,2)+" / "+RIGHT$(FT$(F),2)
5110 PRINTSTRING$( " ",79):RETURN
5150 PRINT " " +F5$(F)+CHR$(4)+" " :GOTO99
5200 INPUT " X$
5205 IFLEFT$(X$,6)="GOTO99" THENRETURN
5210 CC$="" :Y1=CSKV:FORX=0TO79:B$=CHARACTER$(X,Y1-1):IFASC(B$)=0THENB$="" "
5220 CC$=CC$+B$:NEXTX:RETURN
5250 GOSUB7100:INPUT " クリマ ツク (シムラウ No)-(クリマ No) ? - ?
":A$:IFA$="" :I:G=1:GOTO99" THENL=1:M=25:RETURN
5252 IFA$="" GOTO99" THENRETURN
5254 C=LEN(A$):FORI=1TOC:IFMID$(A$,I,1)<>" " THENNEXT

```



```

0 REM
0 REM >>> Auto-2001 CMT-3 <<<
0 REM - UBE Tek Micom Club - S.Nawata
0 REM
1 REW:FORI=1T010:P0KE4544+I,15#I:NEXT
2 CLR:TEMP07:CONSOLE$2,24.C80
3 DIMD$(1,255),I$(1,26),S$(1,26),IC(1,26),IP(1,26),IN(1),IE(1),DE(1),F1$(1),F2$(1),F3$(1),F4$(1),F5$(1),F6$(1),FD$(16),W(26),FT$(1),U$(26),T$(1,26),T(1)
4 DIMM$(26),L$(11),K$(28):IC(1,1)=1
5 GOSUB7200:GOSUB5980:USR($0F14)
6 CURSOR19,21:PRINT"ターゲター? ヲ セットシテ スキ? キー ヲ オシテ フタ? ヲ、"
7 GETA$:IFA$="" THEN7
8 GOSUB6000
10 DEF KEY(1)=セ
11 DEF KEY(2)=フ
12 DEF KEY(3)=カ"メンホ"ー
13 DEF KEY(4)=ハ
14 DEF KEY(5)=ソ
15 DEF KEY(6)=#
16 DEF KEY(7)=ク
17 DEF KEY(8)=シ
18 DEF KEY(9)=テ
19 DEF KEY(10)=GOTO997
21 F=0:G=1
22 GOSUB6100:CURSOR10,1:PRINTSTRING$(" ",23):GOSUB7100:INPUT" ロート" // ファイル No
   =":A$:IFA$="GOTO99" THEN99
24 A=VAL(A$):IF(A>LE)+(A=0) THENGOSUB5000:GOTO22
26 FC$=LEFT$(L$(A),6):Z=A:GOSUB6200:GOSUB6300
83 F1$(F)="" :F2$(F)="" :F3$(F)="" :F4$(F)="" :F5$(F)="" :F6$(F)="" :ROPEN/TF$=GOSUB
7100:PRINT" "+FC$+" ヲ コミテ シュウ" :FORI=1T026:INPUT/TA$,B$,C$:IFA$="" I-END" THEN91
90 I$(F,1)=A$:S$(F,1)=B$:IC(F,1)=VAL(C$):IE(F)=IE(F)+IC(F,1):GOSUB5400:IN(F)=I:
NEXTI
91 FORI=1T0255:INPUT/TD$(F,1):IFD$(F,1)="" D-END" THENINPUT/TFT$(F):DE(F)=I-1:GOTO
93
92 NEXTI
93 FORI=1T026:INPUT/TT$(F,I):IFT$(F,I)="" C-END" THENCLOSE/T:T(F)=I-1:REW:GOTO97
94 NEXTI
97 CURSOR0,24:FORN=0T024:PRINT:FORM=0T030:NEXTM:NEXTN:CONSOLE$0,24:GOSUB7200
99 FD$(11)="" :FD$(14)="" :GOSUB7000:GOSUB5500
111 GOSUB5980
112 C1=1:CURSOR0,0:USR($0F14):INPUT" ":A$:A$=LEFT$(A$,2)
113 IFA$="G" THEN97
120 FORI=1T021STEP2:IFA$=MID$("セウツカ"ハイテ"ソーグシ"SZCYCL",I,2) THEN140
130 NEXTI:GOTO150
140 C2=0:ON(I+1)/2GOTO400,170,175,1000,1500,2000,3000,8000,180,700,190
150 IFC2=1THEN99
155 CURSOR0,2:PRINT"F1=セウツ" F2=メイン" F3=カ"メンホ"ー F4=ハイレフ F5=ロード"
160 CURSOR0,3:PRINT"F6= --- F7=クイラン F8=シ"ト"ウ F9=ターフ F10=メイレイマシ" :GOT
099
170 GOSUB600:GOTO99
175 GOSUB5010:CURSOR0,0:PRINTSPACE$(159):COPY/P1:CONSOLE$0,24:GOTO99
180 W=SIZE:W=(W-560)/(IE(F)+5):W=INT(W):IFW<0THENW=0
182 GOSUB7100:CURSOR0,0:PRINT"ノリサイズ" // フト";W;" キ"ョウ アリマス。 (スキキ"ー ヲ オシテフ
タ"ヲ)"
184 GETA$:IFA$="" THEN184
186 GOTO99
190 GOSUB7100:INPUT" シ"ョフ" ヲ オトリ マスカ (Y/N) " :A$:IFA$<"Y" THEN99
192 CONSOLE$0,24:PRINT" **** シ"ョ"ラク オマセ フタ" ヲ ****"
194 WOPEN/T" ファイル リスト" :PRINT:PRINT" ファイル リスト ヲ カキコミ シュウ"
196 FORI=1T026:PRINT/TL$(I):NEXTI:PRINT/T" ****", "XX", "XX":FORI=1T028:PRINT/TK$(I)
) :NEXTI:CLOSE/T:REW:END
400 CURSOR0,0:PRINT" セウツ" :GOSUB5980
424 GOSUB5250:IFA$="GOTO99" THEN99
43 CONSOLE$2,24:GOSUB7100:PRINT" ニ"ン" キーヲ オシテ ストップ" スグート シマス。 (ストップ" ノト
キ" フ"ト"=セイ、F10=メイレイマシ)" :Y=7:CONSOLE$7,24
427 CURSOR0,3:PRINT" >>> セウツ" <<< "
429 GOSUB5700:FORI=L.TOL+M-1:IFI=DE(F)+1THEN510
430 IF(I=DE(F))*(RIGHT$(D$(F,I),1)="4") THEN510
432 GETA$:IFA$="" THEN438
434 GETA$:IFA$="" THEN434
435 IFA$="" + " THENGOSUB5900:CURSOR0,Y:PRINTSPACE$(79):GOTO450
436 IFASC(A$)=25THEN99
438 GOSUB550:NEXTI:GOTO510
450 IFY=24THENY=Y-1:GOTO434
452 GOTO434
550 IFY=24THENPRINT:Y=Y-1
555 CURSOR0,Y:PRINT" I "+RIGHT$( " "+STR$(I),3)+" "+D$(F,I)
556 IFFD$(11)="" Y THENPRINT/PTAB(0):F6$(F):PRINT/PTAB(0);"} "+RIGHT$( " "+STR$(I
),3)+" "+D$(F,I)
557 Y=Y+1:RETURN
600 IFF=0THENF=1:G=0:RETURN

```

```

610 F=0:G=1:RETURN
700 GOSUB5750
701 GETA$:IFA$="" THEN701
702 IFA$<>"Y" THEN99
709 GOSUB7100:PRINT" *** ファイル コピ - シュウ ***"
710 IFF=0 THENB=1:GOTO730
720 G=0
730 M=DE(G):FORI=1TODE(F):IFSIZE<500 THEN5800
740 D$(G,I)=D$(F,I):NEXTI:FORI=1TO26:IFSIZE<500 THEN5800
750 IC(G,I)=IC(F,I):IP(G,I)=IP(F,I):IF$(G,I)=IF$(F,I):S$(G,I)=S$(F,I):T$(G,I)=T$(F,I):NEXTI
760 DE(G)=DE(F):F1$(G)=F1$(F):F2$(G)=F2$(F):F3$(G)=F3$(F):F4$(G)=F4$(F):F5$(G)=F5$(F):F6$(G)=F6$(F)
770 IN(G)=IN(F):IE(G)=IE(F):FT$(G)=FT$(F):T(G)=T(F)
780 IFDE(G)<M THENFORI=DE(G)+1TOH:D$(G,I)="" :NEXTI
790 GOTO99
1000 E=0:GOSUB7100:GOSUB5070:GOSUB5980:CURSOR0,0:INPUT" ハイリウカ イハイレリウカ イハ CbAdAR
,ツイカ) AbcD+ /ヨツエフ. ハイリウカ ? :$
1075 Z$="" :K=0:FORI=0TO26:W(I)=0:NEXTI:FORI=0TO16:FD$(I)="" :NEXTI
1080 IFA$="" GOTO99 THEN99
1090 A=LEN(A$):GOSUB7100:PRINT" ファイル カキカ マスカ (値/値) ?"
1091 GETY$:IFY$="" THEN1091
1092 IFASC(Y$)=25 THEN99
1100 GOSUB7100:PRINT" インサツ シマスカ (値/値) ?"
1101 GETFD$(11):IFFD$(11)="" THEN1101
1102 IFASC(FD$(11))=25 THEN99
1103 IFFD$(11)="" Y" THENGOSUB5010
1106 E=0:J=1:N=0:FORI=1TO26:CC$=MID$(A$,I,1):IFCC$="" THEN1110
1107 NEXTI
1108 N=1:GOTO1120
1110 IFFD$(14)="" A" THEN1115
1113 GOSUB7100:CURSOR0,0:INPUT" コツエツイカ 日 コツエツイカ ( セン / トキ ) カ コツエツイカ トスル ? "
CC$=IFC$="" GOTO99 THEN99
1114 GOSUB7100:INPUT" クラス ヲ ? :X$:IFX$="" GOTO99 THEN99
1115 D=VAL(X$):D=D+1
1116 L=IN(F)+1:IC(F,L)=D:IFLEFT$(C$,1)="" THENS$(F,L)="" S" :IF(F,L)=RIGHT$(C$,LEN(C$)-1):GOTO1118
1117 S$(F,L)="" N" :IF(F,L)=RIGHT$(SPACE$(40)+C$,IC(F,L)-1):IFFD$(14)="" A" THEN1120
1118 GOSUB7100:INPUT" (オシノモノヲ イレルトキ) ヲルヲ, コツニ イレルトキ) ヲ キーイン) :FD$(B):IFFD$(B)="" GOTO99 THEN99
1120 IFJ=DE(F)+1 THENIFZ$="" THEN1136
1121 IFJ=DE(F)+1 THEN1225
1122 CC$="" :FORI=1TOA:X$=MID$(A$,I,1):IFX$="" THENI=A:GOTO1195
1123 B$=MID$(A$,I,1):IFB$="" THENB=L:M=F:GOTO1140
1124 IFASC(B$)>96 THENB=ASC(B$)-96:M=G:GOTO1132
1130 B=ASC(B$)-64:M=F
1132 Z$=MID$(D$(M,J),IP(M,B),IC(M,B)):CC$=CC$+Z$
1140 IFJ<>1 THEN1195
1150 FD$(3)=FD$(3)+STRING$( "-" ,IC(M,B)-1)+"+"
1160 IF$(M,B)="" N" THENFD$(1)=FD$(1)+RIGHT$(SPACE$(79)+IF$(M,B)+"+",IC(M,B)):GOTO1180
1170 FD$(1)=FD$(1)+LEFT$(IF$(M,B)+SPACE$(79),IC(M,B)-1)+"+"
1180 FD$(2)=FD$(2)+STRING$( "-" ,IC(M,B)-1)+"+" :FD$(4)=FD$(4)+STRING$( "-" ,IC(M,B)-1)+"+" :FD$(5)=FD$(5)+STRING$( "-" ,IC(M,B)-1)+"+"
1185 FD$(6)=FD$(6)+STRING$( "-" ,IC(M,B)-1)+"+"
1187 FD$(10)=FD$(10)+SPACE$(IC(M,B)-1)+"+"
1190 K=K+1:W(K)=IC(M,B):IFY$="" Y" THENH$(K)=IF$(M,B):U$(K)=S$(M,B)
1195 NEXTI:IFJ<>1 THENIFE=1 THEN1310
1197 IFJ<>1 THEN1207
1200 H=0:FORI=1TOK:H=H+W(I):NEXTI:IF(H>74)*$(H<132) THENGOSUB7100:PRINT" 74クワ オ
ハ - シタマス ノヲ ヒョウシテ キマセン." :USR($OF14):E=1:GOTO1304
1202 IFH<131 THEN5800
1203 GOSUB7100:CONSOLE2,24:CURSOR0,3:PRINT" >>> ハイリウ カ イ <<< ( :A$; )"
1204 PRINT" -----+FD$(3)+CHR$(4)+" :PRINT" |No.||"+FD$(1):PRINT" -----+FD$(2)+CH
R$(4)+" :CONSOLE2,24
1205 IFFD$(11)="" Y" THENPRINT/P TAB(0):" -----+FD$(3):PRINT/P TAB(0):" |No.||"+FD$(1)
:PRINT/P TAB(0):" -----+FD$(2)
1207 IFN=1 THEN1220
1208 IFFD$(B)<>" " THENFD$(9)=FD$(B):GOTO1212
1209 USR($OF14):PRINT" | "+RIGHT$( " "+STR$(J),3)+" || "+CC$; :INPUT" :FD$(9):PRIN
TCHR$(2):" :IF(FD$(9)="" GOTO99)+(FD$(9)="" J) THEN99
1212 IF$(M,B)="" N" THENCC$=CC$+RIGHT$(SPACE$(40)+FD$(9)+" | ",0):GOTO1220
1213 CC$=CC$+LEFT$(FD$(9)+SPACE$(40),D-1)+" | "
1220 PRINT" | "+RIGHT$( " "+STR$(J),3)+" || "+CC$; :IFY$="" Y" THEND$(F,J)=CC$
1221 IFFD$(11)="" Y" THENPRINT/P TAB(0):" | "+FD$(10)
1222 IFFD$(11)="" Y" THENPRINT/P TAB(0):" | "+RIGHT$( " "+STR$(J),3)+" || "+CC$; :IFY$="" Y
THEN D$(F,J)=CC$
1224 J=J+1:GOTO1120
1225 PRINT" -----+FD$(6)+CHR$(4)+" :
1226 IFFD$(11)="" Y" THENPRINT/P TAB(0):" -----+FD$(6):FORI=0TO4:PRINT/P NEXTI
1227 IFY$<>"Y" THEN1250
1230 F1$(F)=FD$(3):F2$(F)=FD$(1):F3$(F)=FD$(2):F4$(F)=FD$(5):F5$(F)=FD$(6):IN(F
)=K:IE(F)=0:FORI=1TOK:IF$(F,I)=M$(I):S$(F,I)=U$(I):IC(F,I)=W(I)

```

AUTO-2001 CMT-3 プログラム・リスト

```

1240 IE(F)=IE(F)+IC(F,I):NEXT I:IF(F,I)=1:FOR I=1 TO IN(F)-1:IF(F,I+1)=IP(F,I)+IC(F
,I):NEXT I:IF FD$(14)="A" THEN 2900
1242 GOT099
1250 IF FD$(14)="A" THEN 2900
1252 GOT099
1304 IF FD$(11)<"Y" THEN FOR I=0 TO 9000: NEXT I: GOT099
1305 PRINT/PCHR$(20):TAB(0); " " + FD$(3):PRINT/P TAB(0); " | No. || " + FD$(1):PRINT/P
TAB(0); " " + FD$(2)
1310 IF N=1 THEN 1320
1312 IF FD$(8)<"0" THEN FD$(9)=FD$(8): GOT01316
1314 CONSOLE$2,24:GOSUB 7100:PRINT" カキコ ヲイコ ヲ イレテ クラ ヲイ。 ?":USR(%OF14):PRINT: I
NPUT" : FD$(9): IF (FD$(9)!="GOT099")+(FD$(9)!="J") THEN PRINT/PCHR$(21): GOT099
1316 IF$(M,B)="N" THEN CC%=CC%+RIGHT$(SPACE$(40)+FD$(9)+ " | ",D): GOT01320
1317 CC%=CC%+LEFT$(FD$(9)+SPACE$(40),D-1)+ " | ": GOT01320
1318 PRINT/P TAB(0); " | " + RIGHT$( " " + STR$(J),3)+ " || " + CC%
1320 PRINT/P TAB(0); " | " + FD$(10)
1330 PRINT/P TAB(0); " | " + RIGHT$( " " + STR$(J),3)+ " || " + CC%
1335 J=J+1:Z$=" ": GOT01120
1336 PRINT/P TAB(0); " " + FD$(6)+CHR$(4):FOR I=0 TO 4:PRINT/P: NEXT I:PRINT/PCHR$(21
): GOT099
1500 GOSUB 7100:CURSOR 0,0:PRINT" テーフ " || テーフ ニ セーフ ---- 17 アラニ テーフ ヲリ ロート ---- 27
イル リスト ---- 37 ファイル リスト ---- 47
1510 GETA$=IFA$="" THEN 1510
1520 IF ASC(A$)=25 THEN 99
1530 A=VAL(A$): IF (A>4)+(A=0) THEN GOSUB 5000: GOT01510
1549 IFA=1 THEN 1560
1550 IFA=2 THEN GOSUB 5750: GOT01800
1551 IFA=3 THEN GOSUB 6100: GOT01500
1552 IFA=4 THEN GOSUB 7100: INPUT" リクシ ヲ スル ファイル No. = ":Z$=IF Z$=" GOT099" THEN 99
1553 Z=VAL(Z$): IF (Z>LE)+(Z=0) THEN GOSUB 5000: GOT01552
1554 GOSUB 6230: GOT01500
1560 GOSUB 7100:CURSOR 0,0:PRINT" セーフ " || シラノ ファイル メイト ヒツ ク テー セーフ シマスカ (個/個) ?
":CONSOLE$3,24:CURSOR 0,20:GOSUB 5100
1570 GETA$=IFA$="" THEN 1570
1580 IF ASC(A$)=25 THEN 99
1590 IF (A$!="Y")+(A$!="y") THEN FC%=LEFT$(FT$(F),6): GOT01690
1600 CURSOR 0,12: INPUT" ファイル メイト (6 文字) = ":A$=IFA$=" GOT099" THEN 99
1610 FC%=LEFT$(A$+SPACE$(6),6): INPUT" コメント ( 15 文字) = ":A$=FC%=LEFT$(FC%+
A$+SPACE$(15),21)
1620 INPUT" カギコ 2 ( 2 文字) = ":A$=A$=RIGHT$( " " + A$,2):FC%=FC%+A$
1622 INPUT" カギコ 2 ( 2 文字) = ":A$=A$=RIGHT$( " " + A$,2):FC%=FC%+A$
1624 INPUT" カギコ 2 ( 2 文字) = ":A$=A$=RIGHT$( " " + A$,2):FC%=FC%+A$+FT$(F)=F
C$+L$(LE+1)=FC$:FC%=LEFT$(FC$,6): GOT01690
1630 WOPEN/ T LEFT$(FC$,6):CURSOR 5,22:PRINT LEFT$(FC$,6)+ " カギコ シュウ"
1640 FOR I=1 TO IN(F):PRINT/ T $(F,I),S$(F,I),STR$(IC(F,I)):NEXT I:PRINT/ T " I-END","X
X","XX"
1650 FOR I=1 TO DE(F):PRINT/ T D$(F,I):NEXT I:PRINT/ T " D-END":PRINT/ T FT$(F)
1655 FOR J=1 TO T(F):PRINT/ T T$(F,J):NEXT J:PRINT/ T " C-END":CLOSE/ T:REW:PRINT:PRINT" *
*** シラノ オキ クラ ヲイ ***":USR(%OF14)
1660 GOSUB 6400:WOPEN/ T "ファイル リスト":PRINT:PRINT" ファイル リスト カギコ シュウ"
1665 FOR I=1 TO LE:PRINT/ T L$(I):NEXT I:PRINT/ T " ***","XX","XX":FOR I=1 TO 28:PRINT/ T K$(
I):NEXT I:CLOSE/ T:REW: GOT099
1690 Z=0:FOR I=1 TO LE:IF FC%=LEFT$(L$(I),6) THEN Z=I
1692 NEXT I:IF Z<0 THEN GOT01695
1693 L$(LE+1)=FT$(F):GOSUB 6210:IF LE=0 THEN 99
1694 GOSUB 6300: GOT01630
1695 PRINT:PRINT:PRINT" オナシノ マエノ ファイルカ ステニ アリマスカ":PRINT: INPUT" カギコ カ
(Y/N) ? ":A$=IFA$<"Y" THEN 99
1696 GOSUB 6230: GOT01693
1800 GETB$=IFB$="" THEN 1800
1804 IF B$<"Y" * (B$<"y") THEN 99
1808 GOT022
2000 GOSUB 7100:CURSOR 0,2:GOSUB 5050:CURSOR 0,0:PRINT" ソート ||": INPUT" テーフ カギコ
? ? ":B$=IFB$=" GOT099" THEN 99
2202 I=ASC(B$)-64: IF (I>IN(F))+(I<1) THEN GOSUB 5000: GOT02000
2210 GOSUB 7100:PRINT" カギコ シュン ---- 17 カギコ シュン ---- 27"
2220 GETA$=IFA$="" THEN 2220
2230 IF ASC(A$)=25 THEN 99
2231 GOSUB 7100:PRINT" シラノ シマスカ (個/個) ?"
2232 GETFD$(11):IF FD$(11)="" THEN 2232
2233 IF ASC(FD$(11))=25 THEN 99
2240 A=VAL(A$): IF (A>2)+(A=0) THEN GOSUB 5000: GOT02220
2250 GOSUB 7100:CURSOR 11,0:PRINTB$;" " : IF A=1 THEN PRINT" カギコ ": GOT02270
2260 PRINT" オナシノ "
2270 PRINT" シュンニ カギコ シュウ "
2280 B=IP(F,I):C=IC(F,I):D=I
2282 IF (A=1)*(S$(F,D)!="S") THEN 2320
2284 IF (A=2)*(S$(F,D)!="S") THEN 2370
2285 IF (A=2)*(S$(F,D)!="N") THEN 2350
2290 FOR J=1 TO DE(F)-1:PRINTJ:CHR$(2):A=VAL(MID$(D$(F,J),B,C-1)):W=J:FORK=J+1 TO DE
(F)
2305 IFA=VAL(MID$(D$(F,K),B,C-1)) THEN 2315
2310 A=VAL(MID$(D$(F,K),B,C-1)):W=K

```

```

2315 NEXT: Z$=D$(F,J):D$(F,J)=D$(F,W):D$(F,W)=Z$:NEXT:GOTO2420
2320 FORJ=1TODE(F)-1:PRINTJ:CHR$(2):A$=MID$(D$(F,J),B,C-1):W=J:FORK=J+1TODE(F)
2325 IFA$<MID$(D$(F,K),B,C-1) THEN2335
2330 A$=MID$(D$(F,K),B,C-1):W=K
2335 NEXT: Z$=D$(F,J):D$(F,J)=D$(F,W):D$(F,W)=Z$:NEXT:GOTO2420
2350 FORJ=1TODE(F)-1:PRINTJ:CHR$(2):A$=VAL(MID$(D$(F,J),B,C-1)):W=J:FORK=J+1TODE
(F)
2355 IFA>VAL(MID$(D$(F,K),B,C-1)) THEN2365
2360 A$=VAL(MID$(D$(F,K),B,C-1)):W=K
2365 NEXT: Z$=D$(F,J):D$(F,J)=D$(F,W):D$(F,W)=Z$:NEXT:GOTO2420
2370 FORJ=1TODE(F)-1:PRINTJ:CHR$(2):A$=MID$(D$(F,J),B,C-1):W=J:FORK=J+1TODE(F)
2375 IFA$<MID$(D$(F,K),B,C-1) THEN2390
2380 A$=MID$(D$(F,K),B,C-1):W=K
2390 NEXT: Z$=D$(F,J):D$(F,J)=D$(F,W):D$(F,W)=Z$:NEXT
2420 _CONSOLE0,24:CURSOR0,3:PRINT" >>> ソート <<< { ";I$(F,D);"} " :GOSUBS700:C
ONSOLE7,24
2422 IFFD$(11)="Y" THENPRINT/P TAB(0); " " +F1$(F):PRINT/P TAB(0); " | No. || "+F2$(F)
:PRINT/P TAB(0); " | " +F3$(F)
2430 FORI=1TODE(F):PRINT" | "+RIGHT$( " " +STR$(I),3)+ " || "+D$(F,I)
2431 IFFD$(11)="Y" THENPRINT/P TAB(0); F6$(F):PRINT/P TAB(0); " | "+RIGHT$( " " +STR
$(I),3)+ " || "+D$(F,I)
2432 NEXTI
2435 IFFD$(14)="S" THEN2440
2436 IFFD$(11)="Y" THENPRINT/P TAB(0); " " +F5$(F):FORI=0T04:PRINT/P: NEXT
2437 GOTO5150
2440 PRINT" " +F5$(F)+CHR$(4)+ " || "
2441 IFFD$(11)="Y" THENPRINT/P TAB(0); " " +F5$(F):FORI=0T04:PRINT/P: NEXT
2442 GOTO2900
2900 GOSUB5500: IFFD$(12)="Y" THENFD$(14)="": A$=FD$(13): GOTO8155
2910 GOTO8100
3000 GOSUB7100:CURSOR0,0:PRINT" クイック ソート カン クイック---|| クロ コマンド---|| ヨコ
コマンド---||
3010 GETA$: IFA$="" THEN3010
3020 IFASC(A$)=25 THEN99
3030 A$=VAL(A$): IF(A>3)+(A=0) THENGOSUB5000:GOTO3010
3040 IFA=2 THEN3400
3050 IFA=3 THEN3600
3052 IFF=0 THEN6=1:GOTO3060
3054 F=1:G=0
3060 GOSUBS700:GOSUB7100:INPUT"A+1.2*%B% /ヨウシキライレ(ワカレ キコ) フリ +-%/%" :A$=IFA
$:"GOTO99" THEN99
3070 A$=LEN(A$):GOSUB7100:PRINT" フォルム カキカエ マスカ (値/値) ?"
3080 GETY$:IFY$="" THEN3080
3090 IFASC(Y$)=25 THEN99
3091 GOSUB7100:PRINT" インサツ シマスカ (値/値) ?"
3092 GETFD$(11):IFFD$(11)=" " THEN3092
3093 IFASC(FD$(11))=25 THEN99
3094 IFFD$(11)="Y" THENGOSUB5010
3100 GOSUB7100:INPUT" コマンド メイ ? ":C$=IFC$:"GOTO99" THEN99
3110 GOSUB7100:INPUT" クラ スク ム ? ":X$=IFX$:"GOTO99" THEN99
3115 GOSUB7100:INPUT" ショスツ カ アナタ クラ スク シマスカ ? ":FD$(10):IFFD$(10)=" "GOTO 99"THE
N99
3116 W=VAL(FD$(10)):W=10*W:FD$(10)=STR$(W):W=VAL(FD$(10))
3120 D=VAL(X$):D=D+1:L=IN(F)+1:IC(F,L)=D:S$(F,L)=N":I$(F,L)=C$:J=1
3130 IFJ=DE(F)+1 THEN3960
3131 CC$="" :Q=0:E=0:H=0:N=0:FD$(15)="":FORI=1TOA:B$=MID$(A$,I,1):IFB$="" THENI=
30
3132 FORY=1TO6:IFMID$(" "+$/%^",Y,1)=B$ THENZ$=B$
3134 NEXTY:IF(ASC(B$)=46)+(ASC(B$)>47)*ASC(B$)<58) THENH=1:GOTO3160
3136 IF(ASC(B$)>96)*ASC(B$)<123) THENB=ASC(B$)-96:M=G:GOTO3170
3138 IF(ASC(B$)>64)*ASC(B$)<91) THENB=ASC(B$)-64:M=F:GOTO3170
3150 IF0=1 THENN=VAL(CC$):CC$="" :Q=0:H=0:GOTO3200
3151 IFE=1 THENIFH=1 THENN=-VAL(CC$):CC$="" :H=0:E=0:GOTO3200
3153 IFH=1 THENZ=VAL(CC$):CC$="" :B$=FD$(15):H=0:GOTO3184
3154 IF(I=1)*(Z$="-") THENE=1
3156 GOTO3200
3160 FD$(15)=Z$:IFI=1 THENQ=1
3162 IFI=ATHENZ=VAL(CC$+B$):B$=FD$(15):GOTO3184
3166 CC$=CC$+B$:GOTO3200
3170 Z=VAL(MID$(D$(M,J),IP(M,B),IC(M,B)-1)):IFI=1 THENN=Z:GOTO3200
3171 IFE=1 THENN=-Z:W(49)=0:GOTO3200
3172 B$=Z$:GOTO3184
3184 IFB$="" THENN=N+Z:GOTO3200
3186 IFB$="" THENN=N-Z:GOTO3200
3188 IFB$="" THENN=N+Z:GOTO3200
3190 IFB$="" THENIFZ=0 THENN=0:GOTO3200
3192 IFB$="" THENN=N/Z:GOTO3200
3194 IFB$="" THENIFZ=0 THENN=0:GOTO3200
3195 IFB$="" THENN=N/Z*100:GOTO3200
3197 IFB$="" THENIFN<0 THENN=-N:N=Z:GOTO3200
3198 IFB$="" THENN=N-Z:GOTO3200
3200 NEXTI:N=INT(N*W+0.5):/M:GOSUB3904:GOTO3130
3400 GOSUB7100:CURSOR0,0:PRINT" クラック フォルム カン クイック---|| クロ コマンド---|| ヨコ
コマンド---||

```

AUTO-2001 CMT-3 プログラム・リスト

```

*カエ マスカ (値) ?"
3410 GETY%:IFY%="" THEN$3410
3420 IFASC(Y%)=25THEN99
3421 GOSUB7100:PRINT" インサマスカ (値) ?"
3422 GETFD%(1):IFFD%(1)="" THEN$3422
3423 IFASC(FD%(1))=25THEN99
3424 IFFD%(1)="Y" THENGOSUB5010
3425 GOSUB7100:PRINT" ** クイザン シュウ **"
3430 CONSOLE$2,24:PRINT:PRINT" >>> クアノ コウカイ <<<":Y=7
3440 IFFD%(11)="Y" THENPRINT/PTAB(0); " " + F1%(F):PRINT/PTAB(0); " | No. || + F2%(F)
:PRINT/PTAB(0); " | " + F3%(F)
3441 GOSUB5700:CONSOLE$7,24:FORI=1TODE(F):GOSUB550:NEXTI
3445 C%=LEFT$(F4%(F),LEN(F4%(F))-1)+"4"
3450 PRINT" | " + RIGHT$( " " + STR$(DE(F)+1),3)+" || " + C%
3451 IFFD%(11)="Y" THENPRINT/PTAB(0); F6%(F):PRINT/PTAB(0); " | " + RIGHT$( " " + STR$(
DE(F)+1),3)+" || " + C%
3460 FORI=1TO26:W(I)=0:NEXTI:CC%=LEFT$("クイ" + SPACE$(79),IC(F,1)-1)+" | " :FORJ=2TOI
N(F)
3470 FORI=1TODE(F):W(J)=W(J)+VAL(MID$(D$(F,I),IP(F,J),IC(F,J)-1)):NEXTI
3472 IFB$(F,J)="S" THENCC%=CC%+RIGHT$(SPACE$(79),IC(F,J)-1)+" | " :GOTO3476
3474 CC%=CC%+RIGHT$(SPACE$(79)+STR$(W(J)),IC(F,J)-1)+" | "
3476 NEXTJ
3480 PRINT" | " + RIGHT$( " " + STR$(DE(F)+2),3)+" || " + CC%
3481 IFFD%(11)="Y" THENPRINT/PTAB(0); " | " + RIGHT$( " " + STR$(DE(F)+2),3)+" || " + CC%
3482 IFY%<>"Y" THEN$3492
3490 D%(F,DE(F)+1)=C%:D%(F,DE(F)+2)=CC%:D%(F,DE(F)+3)=C%:DE(F)=DE(F)+3
3492 IFFD%(12)="Y" THENI=VAL(FD%(13)):GOTO3495
3493 IFFD%(11)="Y" THENPRINT/PTAB(0); " | " + F5%(F):FORN=0TO4:PRINT/P:NEXTN
3494 GOTO5150
3495 IFFD%(11)="Y" THENPRINT/PTAB(0); " | " + F5%(F):FORN=0TO4:PRINT/P:NEXTN
3496 PRINT" | " + F5%(F)+CHR$(4)+" "
3497 A%=FD%(13):GOTO8155
3600 GOSUB7100:CURSOR0,0:PRINT"ヨコゴウカイ ファイル カキカエ マスカ (値) ?"
3610 GETY%:IFY%="" THEN$3610
3615 IFASC(Y%)=25THEN99
3616 GOSUB7100:PRINT" インサマスカ (値) ?"
3617 GETFD%(1):IFFD%(1)="" THEN$3617
3618 IFASC(FD%(1))=25THEN99
3619 IFFD%(11)="Y" THENGOSUB5010
3620 GOSUB7100:INPUT" コアノ メイ リ ? ":C%:IFC%="" GOTO999 THEN99
3630 GOSUB7100:INPUT" ク スク リ ? ":X%:IFX%="" GOTO999 THEN99
3640 D=VAL(X%):D=D+1:L=IN(F)+1:IC(F,L)=D:S(F,L)="N":I$(F,L)=C%:A%=""ヨコゴウカイ":
J=1
3650 IFJ=DE(F)+1THEN$3660
3660 N=0:FORI=1TOIN(F):N=N+VAL(MID$(D$(F,I),IP(F,I),IC(F,I)-1)):NEXTI
3670 GOSUB3904:GOTO3650
3904 IFJ<>1THEN$3950
3905 FD%(1)=F1%(F)+STRING$( " ",IC(F,L)-1)+" "
3907 FD%(2)=F2%(F)+RIGHT$(SPACE$(79)+I$(F,L)+" | ",IC(F,L))
3910 FD%(3)=F3%(F)+STRING$( " ",IC(F,L)-1)+" "
3911 FD%(4)=F4%(F)+STRING$( " ",IC(F,L)-1)+" "
3912 FD%(5)=F5%(F)+STRING$( " ",IC(F,L)-1)+" "
3913 IP(F,1)=1:F6%(F)= " | " :FORI=1TOIN(F)-1:IP(F,I+1)=IP(F,I)+IC(F,I):F6%(F)=
F6%(F)+SPACE$(IC(F,I)-1)+" | " :NEXTI
3914 F6%(F)=F6%(F)+SPACE$(IC(F,IN(F))-1)+" | "
3915 FD%(6)=F6%(F)+SPACE$(IC(F,L)-1)+" | "
3940 GOSUB7100:PRINT" ** クイザン シュウ **":CONSOLE$2,24:PRINT
3945 PRINT" >>> クイザン <<< ( " :A%":PRINT" " + FD%(1)+CHR$(4)+" "
3946 IFFD%(11)="Y" THENPRINT/PTAB(0); " " + FD%(1)
3947 PRINT" | No. || + FD%(2):PRINT" | " + FD%(3)+CHR$(4)+"4":CONSOLE$7,24
3948 IFFD%(11)="Y" THENPRINT/PTAB(0); " | No. || + FD%(2):PRINT/PTAB(0); " | " + FD%(3)
3950 X%=D$(F,J)+RIGHT$(SPACE$(40)+STR$(N)+" | ",IC(F,L)):IFY%="" Y" THEND$(F,J)=X%
3952 PRINT" | " + RIGHT$( " " + STR$(J),3)+" || " + X%
3953 IFFD%(11)="Y" THENPRINT/PTAB(0); F6%(6):PRINT/PTAB(0); " | " + RIGHT$( " " + STR$(
J),3)+" || " + X%
3955 J=J+1:RETURN
3960 IFFD%(11)="Y" THENPRINT/PTAB(0); " | " + FD%(5):FORN=0TO4:PRINT/P:NEXTN
3961 PRINT" | " + FD%(5)+CHR$(4)+" " :IFY%="" Y" THENF1%(F)=FD%(1):F2%(F)=FD%(2):F3
%(F)=FD%(3):F4%(F)=FD%(4):F5%(F)=FD%(5):F6%(F)=FD%(6):GOTO3965
3962 IFFD%(12)="Y" Y" THENA%=FD%(13):GOTO8155
3963 GOTO999
3965 IN(F)=IN(F)+1:IE(F)=IE(F)+IC(F,L):IP(F,L)=IP(F,L-1)+IC(F,L-1):IFFD%(12)="Y
" THENA%=FD%(13):GOTO8155
3966 GOTO999
5000 MUSIC" + BOR+BR+E":RETURN
5010 GOSUB7100:PRINT" ** フリクター ラ カワシ ヨシ **":PRINT/PCHR$(17):RETURN
5050 CONSOLE$2,7:PRINT" " + F1%(F):PRINT" | No. || + F2%(F):PRINT" | " + F3%(F):B%
=" " :FORI=1TOIN(F):B%=B%+SPACE$(IC(F,I)-1)+CHR$(64+I):NEXTI
5060 PRINTB%:RETURN
5070 CURSOR68,1:X=73-IE(F):IFX<0THENX=X+1
5071 PRINT" (ノコリ) X:"
5072 CONSOLE$2,10:CURSOR0,2:GOSUB5050:CONSOLE$6,10:CURSOR0,5:IFF=0THENPRINT" -メ
ン-" :GOTO5074

```

```

5073 PRINT"-777-"
5074 CURSOR0,6:PRINT"      "+F1$(G):PRINT" |No.|" +F2$(G):PRINT" |      "+F3$(G):B$="
      :FORI=1TOIN(G):B$=B$+SPACE$(IC(G,I)-1)+CHR$(96+I):NEXTI
5076 PRINTB$:CURSOR0,9:IFG=0THENPRINT"-メイン-" :RETURN
5078 PRINT"-777-" :RETURN
5100 PRINT"Title = " :LEFT$(FT$(F),6):" " :MID$(FT$(F),7,15):TAB(64):"Date = " :M
ID$(FT$(F),22,2)+"/"+MID$(FT$(F),24,2)+"/"+RIGHT$(FT$(F),2)
5110 PRINTSTRING$(" ",79):RETURN
5150 PRINT"      "+F5$(F)+CHR$(4)+"J " :GOTO99
5200 INPUT" " :X$
5205 IFLEFT$(X$,6)="GOTO99" THENRETURN
5210 C$="" :Y1=CSRV:FORX=0TO79:B$=CHARACTER$(X,Y1-1):IFASC(B$)=0THENB$=""
5220 C$=C$+B$:NEXTI:RETURN
5250 GOSUB7100:INPUT"      "+F1$+F2$+F3$+F4$+F5$+F6$+F7$+F8$+F9$+F10$+F11$+F12$+F13$+F14$+F15$+F16$+F17$+F18$+F19$+F20$+F21$+F22$+F23$+F24$+F25$+F26$+F27$+F28$+F29$+F30$+F31$+F32$+F33$+F34$+F35$+F36$+F37$+F38$+F39$+F40$+F41$+F42$+F43$+F44$+F45$+F46$+F47$+F48$+F49$+F50$+F51$+F52$+F53$+F54$+F55$+F56$+F57$+F58$+F59$+F60$+F61$+F62$+F63$+F64$+F65$+F66$+F67$+F68$+F69$+F70$+F71$+F72$+F73$+F74$+F75$+F76$+F77$+F78$+F79$+F80$+F81$+F82$+F83$+F84$+F85$+F86$+F87$+F88$+F89$+F90$+F91$+F92$+F93$+F94$+F95$+F96$+F97$+F98$+F99$+F100$+F101$+F102$+F103$+F104$+F105$+F106$+F107$+F108$+F109$+F110$+F111$+F112$+F113$+F114$+F115$+F116$+F117$+F118$+F119$+F120$+F121$+F122$+F123$+F124$+F125$+F126$+F127$+F128$+F129$+F130$+F131$+F132$+F133$+F134$+F135$+F136$+F137$+F138$+F139$+F140$+F141$+F142$+F143$+F144$+F145$+F146$+F147$+F148$+F149$+F150$+F151$+F152$+F153$+F154$+F155$+F156$+F157$+F158$+F159$+F160$+F161$+F162$+F163$+F164$+F165$+F166$+F167$+F168$+F169$+F170$+F171$+F172$+F173$+F174$+F175$+F176$+F177$+F178$+F179$+F180$+F181$+F182$+F183$+F184$+F185$+F186$+F187$+F188$+F189$+F190$+F191$+F192$+F193$+F194$+F195$+F196$+F197$+F198$+F199$+F200$+F201$+F202$+F203$+F204$+F205$+F206$+F207$+F208$+F209$+F210$+F211$+F212$+F213$+F214$+F215$+F216$+F217$+F218$+F219$+F220$+F221$+F222$+F223$+F224$+F225$+F226$+F227$+F228$+F229$+F230$+F231$+F232$+F233$+F234$+F235$+F236$+F237$+F238$+F239$+F240$+F241$+F242$+F243$+F244$+F245$+F246$+F247$+F248$+F249$+F250$+F251$+F252$+F253$+F254$+F255$+F256$+F257$+F258$+F259$+F260$+F261$+F262$+F263$+F264$+F265$+F266$+F267$+F268$+F269$+F270$+F271$+F272$+F273$+F274$+F275$+F276$+F277$+F278$+F279$+F280$+F281$+F282$+F283$+F284$+F285$+F286$+F287$+F288$+F289$+F290$+F291$+F292$+F293$+F294$+F295$+F296$+F297$+F298$+F299$+F300$+F301$+F302$+F303$+F304$+F305$+F306$+F307$+F308$+F309$+F310$+F311$+F312$+F313$+F314$+F315$+F316$+F317$+F318$+F319$+F320$+F321$+F322$+F323$+F324$+F325$+F326$+F327$+F328$+F329$+F330$+F331$+F332$+F333$+F334$+F335$+F336$+F337$+F338$+F339$+F340$+F341$+F342$+F343$+F344$+F345$+F346$+F347$+F348$+F349$+F350$+F351$+F352$+F353$+F354$+F355$+F356$+F357$+F358$+F359$+F360$+F361$+F362$+F363$+F364$+F365$+F366$+F367$+F368$+F369$+F370$+F371$+F372$+F373$+F374$+F375$+F376$+F377$+F378$+F379$+F380$+F381$+F382$+F383$+F384$+F385$+F386$+F387$+F388$+F389$+F390$+F391$+F392$+F393$+F394$+F395$+F396$+F397$+F398$+F399$+F400$+F401$+F402$+F403$+F404$+F405$+F406$+F407$+F408$+F409$+F410$+F411$+F412$+F413$+F414$+F415$+F416$+F417$+F418$+F419$+F420$+F421$+F422$+F423$+F424$+F425$+F426$+F427$+F428$+F429$+F430$+F431$+F432$+F433$+F434$+F435$+F436$+F437$+F438$+F439$+F440$+F441$+F442$+F443$+F444$+F445$+F446$+F447$+F448$+F449$+F450$+F451$+F452$+F453$+F454$+F455$+F456$+F457$+F458$+F459$+F460$+F461$+F462$+F463$+F464$+F465$+F466$+F467$+F468$+F469$+F470$+F471$+F472$+F473$+F474$+F475$+F476$+F477$+F478$+F479$+F480$+F481$+F482$+F483$+F484$+F485$+F486$+F487$+F488$+F489$+F490$+F491$+F492$+F493$+F494$+F495$+F496$+F497$+F498$+F499$+F500$+F501$+F502$+F503$+F504$+F505$+F506$+F507$+F508$+F509$+F510$+F511$+F512$+F513$+F514$+F515$+F516$+F517$+F518$+F519$+F520$+F521$+F522$+F523$+F524$+F525$+F526$+F527$+F528$+F529$+F530$+F531$+F532$+F533$+F534$+F535$+F536$+F537$+F538$+F539$+F540$+F541$+F542$+F543$+F544$+F545$+F546$+F547$+F548$+F549$+F550$+F551$+F552$+F553$+F554$+F555$+F556$+F557$+F558$+F559$+F560$+F561$+F562$+F563$+F564$+F565$+F566$+F567$+F568$+F569$+F570$+F571$+F572$+F573$+F574$+F575$+F576$+F577$+F578$+F579$+F580$+F581$+F582$+F583$+F584$+F585$+F586$+F587$+F588$+F589$+F590$+F591$+F592$+F593$+F594$+F595$+F596$+F597$+F598$+F599$+F600$+F601$+F602$+F603$+F604$+F605$+F606$+F607$+F608$+F609$+F610$+F611$+F612$+F613$+F614$+F615$+F616$+F617$+F618$+F619$+F620$+F621$+F622$+F623$+F624$+F625$+F626$+F627$+F628$+F629$+F630$+F631$+F632$+F633$+F634$+F635$+F636$+F637$+F638$+F639$+F640$+F641$+F642$+F643$+F644$+F645$+F646$+F647$+F648$+F649$+F650$+F651$+F652$+F653$+F654$+F655$+F656$+F657$+F658$+F659$+F660$+F661$+F662$+F663$+F664$+F665$+F666$+F667$+F668$+F669$+F670$+F671$+F672$+F673$+F674$+F675$+F676$+F677$+F678$+F679$+F680$+F681$+F682$+F683$+F684$+F685$+F686$+F687$+F688$+F689$+F690$+F691$+F692$+F693$+F694$+F695$+F696$+F697$+F698$+F699$+F700$+F701$+F702$+F703$+F704$+F705$+F706$+F707$+F708$+F709$+F710$+F711$+F712$+F713$+F714$+F715$+F716$+F717$+F718$+F719$+F720$+F721$+F722$+F723$+F724$+F725$+F726$+F727$+F728$+F729$+F730$+F731$+F732$+F733$+F734$+F735$+F736$+F737$+F738$+F739$+F740$+F741$+F742$+F743$+F744$+F745$+F746$+F747$+F748$+F749$+F750$+F751$+F752$+F753$+F754$+F755$+F756$+F757$+F758$+F759$+F760$+F761$+F762$+F763$+F764$+F765$+F766$+F767$+F768$+F769$+F770$+F771$+F772$+F773$+F774$+F775$+F776$+F777$+F778$+F779$+F780$+F781$+F782$+F783$+F784$+F785$+F786$+F787$+F788$+F789$+F790$+F791$+F792$+F793$+F794$+F795$+F796$+F797$+F798$+F799$+F800$+F801$+F802$+F803$+F804$+F805$+F806$+F807$+F808$+F809$+F810$+F811$+F812$+F813$+F814$+F815$+F816$+F817$+F818$+F819$+F820$+F821$+F822$+F823$+F824$+F825$+F826$+F827$+F828$+F829$+F830$+F831$+F832$+F833$+F834$+F835$+F836$+F837$+F838$+F839$+F840$+F841$+F842$+F843$+F844$+F845$+F846$+F847$+F848$+F849$+F850$+F851$+F852$+F853$+F854$+F855$+F856$+F857$+F858$+F859$+F860$+F861$+F862$+F863$+F864$+F865$+F866$+F867$+F868$+F869$+F870$+F871$+F872$+F873$+F874$+F875$+F876$+F877$+F878$+F879$+F880$+F881$+F882$+F883$+F884$+F885$+F886$+F887$+F888$+F889$+F890$+F891$+F892$+F893$+F894$+F895$+F896$+F897$+F898$+F899$+F900$+F901$+F902$+F903$+F904$+F905$+F906$+F907$+F908$+F909$+F910$+F911$+F912$+F913$+F914$+F915$+F916$+F917$+F918$+F919$+F920$+F921$+F922$+F923$+F924$+F925$+F926$+F927$+F928$+F929$+F930$+F931$+F932$+F933$+F934$+F935$+F936$+F937$+F938$+F939$+F940$+F941$+F942$+F943$+F944$+F945$+F946$+F947$+F948$+F949$+F950$+F951$+F952$+F953$+F954$+F955$+F956$+F957$+F958$+F959$+F960$+F961$+F962$+F963$+F964$+F965$+F966$+F967$+F968$+F969$+F970$+F971$+F972$+F973$+F974$+F975$+F976$+F977$+F978$+F979$+F980$+F981$+F982$+F983$+F984$+F985$+F986$+F987$+F988$+F989$+F990$+F991$+F992$+F993$+F994$+F995$+F996$+F997$+F998$+F999$+F1000$+F1001$+F1002$+F1003$+F1004$+F1005$+F1006$+F1007$+F1008$+F1009$+F1010$+F1011$+F1012$+F1013$+F1014$+F1015$+F1016$+F1017$+F1018$+F1019$+F1020$+F1021$+F1022$+F1023$+F1024$+F1025$+F1026$+F1027$+F1028$+F1029$+F1030$+F1031$+F1032$+F1033$+F1034$+F1035$+F1036$+F1037$+F1038$+F1039$+F1040$+F1041$+F1042$+F1043$+F1044$+F1045$+F1046$+F1047$+F1048$+F1049$+F1050$+F1051$+F1052$+F1053$+F1054$+F1055$+F1056$+F1057$+F1058$+F1059$+F1060$+F1061$+F1062$+F1063$+F1064$+F1065$+F1066$+F1067$+F1068$+F1069$+F1070$+F1071$+F1072$+F1073$+F1074$+F1075$+F1076$+F1077$+F1078$+F1079$+F1080$+F1081$+F1082$+F1083$+F1084$+F1085$+F1086$+F1087$+F1088$+F1089$+F1090$+F1091$+F1092$+F1093$+F1094$+F1095$+F1096$+F1097$+F1098$+F1099$+F1100$+F1101$+F1102$+F1103$+F1104$+F1105$+F1106$+F1107$+F1108$+F1109$+F1110$+F1111$+F1112$+F1113$+F1114$+F1115$+F1116$+F1117$+F1118$+F1119$+F1120$+F1121$+F1122$+F1123$+F1124$+F1125$+F1126$+F1127$+F1128$+F1129$+F1130$+F1131$+F1132$+F1133$+F1134$+F1135$+F1136$+F1137$+F1138$+F1139$+F1140$+F1141$+F1142$+F1143$+F1144$+F1145$+F1146$+F1147$+F1148$+F1149$+F1150$+F1151$+F1152$+F1153$+F1154$+F1155$+F1156$+F1157$+F1158$+F1159$+F1160$+F1161$+F1162$+F1163$+F1164$+F1165$+F1166$+F1167$+F1168$+F1169$+F1170$+F1171$+F1172$+F1173$+F1174$+F1175$+F1176$+F1177$+F1178$+F1179$+F1180$+F1181$+F1182$+F1183$+F1184$+F1185$+F1186$+F1187$+F1188$+F1189$+F1190$+F1191$+F1192$+F1193$+F1194$+F1195$+F1196$+F1197$+F1198$+F1199$+F1200$+F1201$+F1202$+F1203$+F1204$+F1205$+F1206$+F1207$+F1208$+F1209$+F1210$+F1211$+F1212$+F1213$+F1214$+F1215$+F1216$+F1217$+F1218$+F1219$+F1220$+F1221$+F1222$+F1223$+F1224$+F1225$+F1226$+F1227$+F1228$+F1229$+F1230$+F1231$+F1232$+F1233$+F1234$+F1235$+F1236$+F1237$+F1238$+F1239$+F1240$+F1241$+F1242$+F1243$+F1244$+F1245$+F1246$+F1247$+F1248$+F1249$+F1250$+F1251$+F1252$+F1253$+F1254$+F1255$+F1256$+F1257$+F1258$+F1259$+F1260$+F1261$+F1262$+F1263$+F1264$+F1265$+F1266$+F1267$+F1268$+F1269$+F1270$+F1271$+F1272$+F1273$+F1274$+F1275$+F1276$+F1277$+F1278$+F1279$+F1280$+F1281$+F1282$+F1283$+F1284$+F1285$+F1286$+F1287$+F1288$+F1289$+F1290$+F1291$+F1292$+F1293$+F1294$+F1295$+F1296$+F1297$+F1298$+F1299$+F1300$+F1301$+F1302$+F1303$+F1304$+F1305$+F1306$+F1307$+F1308$+F1309$+F1310$+F1311$+F1312$+F1313$+F1314$+F1315$+F1316$+F1317$+F1318$+F1319$+F1320$+F1321$+F1322$+F1323$+F1324$+F1325$+F1326$+F1327$+F1328$+F1329$+F1330$+F1331$+F1332$+F1333$+F1334$+F1335$+F1336$+F1337$+F1338$+F1339$+F1340$+F1341$+F1342$+F1343$+F1344$+F1345$+F1346$+F1347$+F1348$+F1349$+F1350$+F1351$+F1352$+F1353$+F1354$+F1355$+F1356$+F1357$+F1358$+F1359$+F1360$+F1361$+F1362$+F1363$+F1364$+F1365$+F1366$+F1367$+F1368$+F1369$+F1370$+F1371$+F1372$+F1373$+F1374$+F1375$+F1376$+F1377$+F1378$+F1379$+F1380$+F1381$+F1382$+F1383$+F1384$+F1385$+F1386$+F1387$+F1388$+F1389$+F1390$+F1391$+F1392$+F1393$+F1394$+F1395$+F1396$+F1397$+F1398$+F1399$+F1400$+F1401$+F1402$+F1403$+F1404$+F1405$+F1406$+F1407$+F1408$+F1409$+F1410$+F1411$+F1412$+F1413$+F1414$+F1415$+F1416$+F1417$+F1418$+F1419$+F1420$+F1421$+F1422$+F1423$+F1424$+F1425$+F1426$+F1427$+F1428$+F1429$+F1430$+F1431$+F1432$+F1433$+F1434$+F1435$+F1436$+F1437$+F1438$+F1439$+F1440$+F1441$+F1442$+F1443$+F1444$+F1445$+F1446$+F1447$+F1448$+F1449$+F1450$+F1451$+F1452$+F1453$+F1454$+F1455$+F1456$+F1457$+F1458$+F1459$+F1460$+F1461$+F1462$+F1463$+F1464$+F1465$+F1466$+F1467$+F1468$+F1469$+F1470$+F1471$+F1472$+F1473$+F1474$+F1475$+F1476$+F1477$+F1478$+F1479$+F1480$+F1481$+F1482$+F1483$+F1484$+F1485$+F1486$+F1487$+F1488$+F1489$+F1490$+F1491$+F1492$+F1493$+F1494$+F1495$+F1496$+F1497$+F1498$+F1499$+F1500$+F1501$+F1502$+F1503$+F1504$+F1505$+F1506$+F1507$+F1508$+F1509$+F1510$+F1511$+F1512$+F1513$+F1514$+F1515$+F1516$+F1517$+F1518$+F1519$+F1520$+F1521$+F1522$+F1523$+F1524$+F1525$+F1526$+F1527$+F1528$+F1529$+F1530$+F1531$+F1532$+F1533$+F1534$+F1535$+F1536$+F1537$+F1538$+F1539$+F1540$+F1541$+F1542$+F1543$+F1544$+F1545$+F1546$+F1547$+F1548$+F1549$+F1550$+F1551$+F1552$+F1553$+F1554$+F1555$+F1556$+F1557$+F1558$+F1559$+F1560$+F1561$+F1562$+F1563$+F1564$+F1565$+F1566$+F1567$+F1568$+F1569$+F1570$+F1571$+F1572$+F1573$+F1574$+F1575$+F1576$+F1577$+F1578$+F1579$+F1580$+F1581$+F1582$+F1583$+F1584$+F1585$+F1586$+F1587$+F1588$+F1589$+F1590$+F1591$+F1592$+F1593$+F1594$+F1595$+F1596$+F1597$+F1598$+F1599$+F1600$+F1601$+F1602$+F1603$+F1604$+F1605$+F1606$+F1607$+F1608$+F1609$+F1610$+F1611$+F1612$+F1613$+F1614$+F1615$+F1616$+F1617$+F1618$+F1619$+F1620$+F1621$+F1622$+F1623$+F1624$+F1625$+F1626$+F1627$+F1628$+F1629$+F1630$+F1631$+F1632$+F1633$+F1634$+F1635$+F1636$+F1637$+F1638$+F1639$+F1640$+F1641$+F1642$+F1643$+F1644$+F1645$+F1646$+F1647$+F1648$+F1649$+F1650$+F1651$+F1652$+F1653$+F1654$+F1655$+F1656$+F1657$+F1658$+F1659$+F1660$+F1661$+F1662$+F1663$+F1664$+F1665$+F1666$+F1667$+F1668$+F1669$+F1670$+F1671$+F1672$+F1673$+F1674$+F1675$+F1676$+F1677$+F1678$+F1679$+F1680$+F1681$+F1682$+F1683$+F1684$+F1685$+F1686$+F1687$+F1688$+F1689$+F1690$+F1691$+F1692$+F1693$+F1694$+F1695$+F1696$+F1697$+F1698$+F1699$+F1700$+F1701$+F1702$+F1703$+F1704$+F1705$+F1706$+F1707$+F1708$+F1709$+F1710$+F1711$+F1712$+F1713$+F1714$+F1715$+F1716$+F1717$+F1718$+F1719$+F1720$+F1721$+F1722$+F1723$+F1724$+F1725$+F1726$+F1727$+F1728$+F1729$+F1730$+F1731$+F1732$+F1733$+F1734$+F1735$+F1736$+F1737$+F1738$+F1739$+F1740$+F1741$+F1742$+F1743$+F1744$+F1745$+F1746$+F1747$+F1748$+F1749$+F1750$+F1751$+F1752$+F1753$+F1754$+F1755$+F1756$+F1757$+F1758$+F1759$+F1760$+F1761$+F1762$+F1763$+F1764$+F1765$+F1766$+F1767$+F1768$+F1769$+F1770$+F1771$+F1772$+F1773$+F1774$+F1775$+F1776$+F1777$+F1778$+F1779$+F1780$+F1781$+F1782$+F1783$+F1784$+F1785$+F1786$+F1787$+F1788$+F1789$+F1790$+F1791$+F1792$+F1793$+F1794$+F1795$+F1796$+F1797$+F1798$+F1799$+F1800$+F1801$+F1802$+F1803$+F1804$+F1805$+F1806$+F1807$+F1808$+F1809$+F1810$+F1811$+F1812$+F1813$+F1814$+F1815$+F1816$+F1817$+F1818$+F1819$+F1820$+F1821$+F1822$+F1823$+F1824$+F1825$+F1826$+F1827$+F1828$+F1829$+F1830$+F1831$+F1832$+F1833$+F1834$+F1835$+F1836$+F1837$+F1838$+F1839$+F1840$+F1841$+F1842$+F1843$+F1844$+F1845$+F1846$+F1847$+F1848$+F1849$+F1850$+F1851$+F1852$+F1853$+F1854$+F1855$+F1856$+F1857$+F1858$+F1859$+F1860$+F1861$+F186
```

AUTO-2001 CMT-3 プログラム・リスト

```

6310 OUT@227,13:OUT@224,24:OUT@224,16:RETURN
6400 FORI=0TO3000:NEXT
6405 J=200
6410 INP@225,X:IFX>1996GOTO6405
6415 J=J-1:IFJ>0THEN6410
6420 RETURN
7000 CURSOR0,0:PRINTSPACE*(79):CURSOR0,0:PRINT"★メイレマ★" Ⅱ ツカヌ メイレ = F1~F10,
S2=コリ ザズ, CY=ファミル コヒー, CL=ジ ョ"ラ オウ"
7010 CURSOR0,1:PRINTSTRING(" ",79):IFC2=1THENCURSOR0,2:PRINTSPACE*(79):CURSOR
,3:PRINTSPACE*(77):C2=0:RETURN
7020 C2=1:RETURN
7100 CURSOR0,1:PRINTSTRING(" ",79):CURSOR9,0:PRINTSPACE*(70):CURSOR9,0:RETURN
7200 CURSOR26,8:PRINT">>> Auto-2001 CMT-3 <<<"CURSOR24,11:PRINT"*** Ube Tek M
icom Club ***":RETURN
8000 GOSUB7100:CURSOR0,0:PRINT" シ"ト"ウ Ⅱ ★★ フ"ロ"ラム リスト ★★"
8010 CONSOLE2,24:CURSOR0,3:PRINT"No. S フ"ロ"ラムメイ フ"ロ"ラム ノ シキ
W ヲモクメイ CH . P C No.":PRINTSTRING(" ",75):Y=5:CONSOLE5,24
8020 FORI=1TOT(F):IFT(F)=0THEN8060
8025 GETA$:IFA$="" THEN8030
8026 GETA$:IFA$="" THEN8026
8028 IFASC(A$)=25THEN8100
8030 A$=LEFT$(STR$(I)+" ",2)+"Ⅱ":PRINTA$:CURSOR4,Y:PRINTT$(F,I)
8050 Y=Y+1:NEXTI
8060 PRINTSTRING(" ",75)
8100 USR(FOF14):GOSUB7100:PRINT" フ"ロ"ラムノ シ"ヨコウ---Ⅱ フ"ロノ トフク---Ⅱ フ"ロノ テイセイ---
Ⅱ フ"ロ"ラムリスト---Ⅱ"
8110 GETA$:IFA$="" THEN8110
8120 IFASC(A$)=25THEN99
8130 A=VAL(A$):IF(A>4)+(A=0)THENGOSUB5000:GOTO8110
8135 IFA=4THEN8000
8140 IFA=2THEN8200
8150 IFA=3THEN8400
8154 GOSUB7100:INPUT" フ"ロ"ラム No. = ":A$:IFA$="GOTO99" THEN99
8155 I=VAL(A$):A$=MID$(T$(0,I),15,30):Y$=MID$(T$(0,I),46,1):C$=MID$(T$(0,I),48,
10):X$=MID$(T$(0,I),59,2):FD$(10)=MID$(T$(0,I),62,1):FD$(11)=MID$(T$(0,I),64,1)
8156 FD$(12)=MID$(T$(0,I),66,1):FD$(13)=RIGHT$(T$(0,I),2):FD$(14)=LEFT$(T$(0,I)
,1):IFFD$(11)="Y" THENGOSUB5010
8157 IFFD$(14)="I" THENA=30:GOTO3116
8158 IFFD$(14)="C" THENA25
8159 IFFD$(14)="L" THENA640
8160 IFFD$(14)="A" THENZ$="":A=30:K=0:FORI=1TO10:FD$(I)="":NEXTI:FD$(8)="@" :FORI
=1TO26:W(I)=0:NEXTI:GOTO1106
8161 IFFD$(14)="S" THENB$=LEFT$(A$,1):A$=MID$(A$,2,1):I=ASC(B$)-64:GOTO2240
8162 IFFD$(14)="X" THENX=F:F=G:G=X:A$=FD$(13):GOTO8155
8163 GOSUB5300:GOTO99
8200 CURSOR0,Y:PRINTSPACE*(79):CURSOR0,Y:FORI=Y-4TO26:A$=LEFT$(STR$(I)+" ",2)+
"Ⅱ":PRINTA$
8210 USR(FOF14):GOSUB7100:CURSOR0,0:INPUT" ト"ロ"ク Ⅱ]=オウリ. フ"ロ"ラムノ シヨル (クイケン=I
,ツカヌ=C,ヨウリ=L,ハイレウ=A,ソート=S,X=チェンジ) ":A$:IFA$="3" THENT(F)=I-1:GOTO8100
8218 T$(F,I)=LEFT$(A$+" ",1)+" ":CURSOR4,Y:PRINTA$
8220 GOSUB7100:INPUT" フ"ロ"ラムノ ナメ (10%) ":A$
8228 T$(F,I)=T$(F,I)+LEFT$(A$+SPACE$(10),10)+" ":CURSOR6,Y:PRINTA$
8230 GOSUB7100:INPUT" フ"ロ"ラムノ シキ (30%) ":A$
8238 T$(F,I)=T$(F,I)+LEFT$(A$+SPACE$(30),30)+" ":CURSOR18,Y:PRINTA$
8240 GOSUB7100:INPUT" フ"ロ"ラムノ カネ (Y/N) ":A$
8248 T$(F,I)=T$(F,I)+LEFT$(A$+" ",1)+" ":CURSOR49,Y:PRINTA$
8250 GOSUB7100:INPUT" ヲモクメイ (10%) ":A$
8258 T$(F,I)=T$(F,I)+RIGHT$(SPACE$(10)+A$,10)+" ":CURSOR51,Y:PRINTA$
8260 GOSUB7100:INPUT" ヲモク ":A$
8268 T$(F,I)=T$(F,I)+LEFT$(A$+" ",2)+" ":CURSOR62,Y:PRINTA$
8270 GOSUB7100:INPUT" ショウグ"イ"ン"ク"ラ"ヲ"シマスカ ":A$
8278 T$(F,I)=T$(F,I)+LEFT$(A$+" ",1)+" ":CURSOR65,Y:PRINTA$
8280 GOSUB7100:INPUT" イ"ン"ク"ラ"シマスカ (Y/N) ":A$
8288 T$(F,I)=T$(F,I)+LEFT$(A$+" ",1)+" ":CURSOR67,Y:PRINTA$
8290 GOSUB7100:INPUT" レ"ン"ク"ク"イ"ン"ク"ラ"シマスカ (Y/N) ":A$
8298 T$(F,I)=T$(F,I)+LEFT$(A$+" ",1)+" ":CURSOR69,Y:PRINTA$
8300 GOSUB7100:INPUT" ツ"カ"ル" フ"ロ"ラム No. ":A$
8308 T$(F,I)=T$(F,I)+LEFT$(A$+" ",2):CURSOR71,Y:PRINTA$:Y=Y+1:NEXTI
8400 GOSUB7100:INPUT" (タイ"ロ"ED) ヲモク"シ"ル" フ"ロ"ラム No. = ":A$
8405 IFA$="ED" THEN8500
8410 IFA$="GOTO99" THEN99
8420 A=VAL(A$):IFA=0THEN8400
8425 GOSUB5750
8426 GETA$:IFA$="" THEN8426
8427 IFA$(">")Y" THEN8100
8430 FORI=AT025:T$(F,I)=T$(F,I+1):NEXTI:T(F)=T(F)-1:GOTO8100
8500 GOSUB7100:CURSOR0,0:PRINT"ED Ⅱ タイ"ロ"ハ"1"キ"ヨウ"コ"ト"ニ"オカ"リ"マス."
8510 CURSOR0,0:USR(FOF14):INPUT"IC$
8520 IFLEFT$(C$,6)="GOTO99" THEN8000
8530 IFMID$(C$,3,1)="Ⅱ" THENC=VAL(LEFT$(C$,2)):GOTO8550
8540 GOTO8000
8550 T$(F,C)=MID$(C$,5,69):GOTO8500

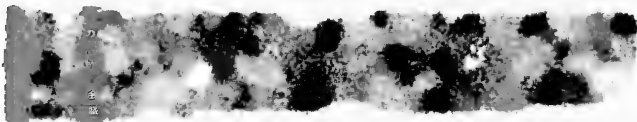
```




AUTO-2001 CMT-2

& TOSの設計について

宇部鉄工所マイコンクラブ 縄田 悟
奥田伸也



CMT-2にはG-RAM1が必要です。このグラフィックとキャラクタを同時コピーするためにI/O 82年4月号に藤原秀樹氏により紹介されたプログラムをSF000にリロケートして使わせていただいています。

特 長

オプション機能として選んだものは、

- ①小計計算(ソート小計機能付)
- ②集計機能(カウンタ機能付)
- ③テーブル検索(サイジング検索付)
- ④自由計算(演算結果の自由書き込み機能付)

また、グラフ機能として選んだものは

- ①棒グラフ(数値表示とスクリーン・メッシュ付)
- ②折れ線グラフ(スクリーン・メッシュ付)
- ③円グラフ(%表示付)
- ④帯グラフ(%表示付)
- ⑤バレット図(ソート機能付)

というふうにずいぶん盛り込みました。そのためデータ・エリアがやや不足気味で、16K(80桁で200行)以上のデータ・ファイルは、あらかじめ不要の部分を削除してCMT-2にかけてください。

CMT-2の使用方法

CMT-1およびCMT-3と共通の機能の説明は省略しますが、このパッケージへのデータ・ファイル・アクセスはCMT-1で作ったTOS管理下のデータ・ファイルとしているので4月、5月号を参照してください。

①CMT-2をLOAD RUNするとCMT-2本体のすぐ後にセーブしてある「COPY SF000」をLOADしてそのREWを行ない、REW完了後「データテープヲセットシテスキナキヲオシテクダサイ」とメッセージが出るので、データ・テープ(CMT-1のTOSで作ったテープ)をセットしてキーを押すと、ファイル・リストを読み込み、ファイル・リストを表示して、ファイルNo.を要求してきます。

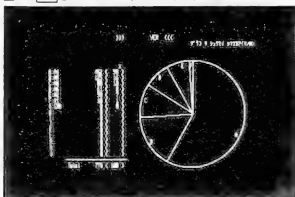
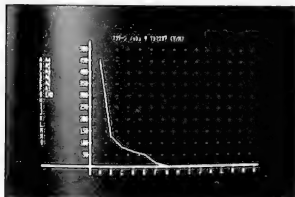
②希望のファイルNo.をキーインするとファイルを読み込んで命令待ちになります。

ここでCMT-2で使える命令表は「CR」キーを押すことにより表示してきますが、

F1=ヒョウ	F2=メイングラフ	F3=カ・メソコト	F4=---	F5=---
F6=ワ・ラフ	F7=オ・ラ・ジョ	F8=---	F9=テープ	F10=マイレバメキ

の8種類です。

③まず「F6」をキーインしてみましょう。すると、グラフに



する項目とグラフにする行番号（グラフにできるデータ数は15までとしています。それ以上は無視されます）。フルスケールの値を次々に聞いてくるので、それらをキーインします。

4画面がクリアされて“タンイ ハ?”と聞いてきますが、これはグラフの目盛に入れる単位名ですから適当に、たとえば“マンエン”とか“km”というふうに入れてください。

5次にグラフの種類を聞いてくるので、まず1をキーインして棒グラフを作ることにしましょう。すると“スウチ ヲ ダシマスカ?”と聞いてきますが、これは棒グラフの横にそのデータの数値を表示するか、という意味でYをキーインすると図1のような表示になります。

6すべてのグラフ表示は第1項目を各データのタイトルとし、グラフ全体のタイトルは選んだ項目名とするので注意してください。

7棒グラフを表示した後“スクリーン メッシュ ヲ ダシマスカ?”と聞いてきます。Yとすると図2のようなスクリーン・メッシュを入れます。Nとすると“グラフ ノ シュルイ ヲ カエマスカ? F10=メイレイマチ”と聞いてきます。Yとすればグラフの選択に戻ります。Nとすると“プリント シマスカ?”と聞いてきます。Yとすれば画面をそっくりプリントします。

8では次に折れ線グラフにしましょう（グラフの選択の前に“データヲオオキイジュニナラベカエマスカ?”と聞いてきます。円グラフや帯グラフ、特にバレート図にする前はYとすると大きい順に並べ換えます）。2をキーインするとただちに図3のように表示し、スクリーン・メッシュやプリントを聞いてくるのは棒グラフと同じです。

9今度は、大きい順に並べ換えた後、円グラフにしてみましょう。3をキーインすると図4のような円グラフを表示します。この場合は%表示とTotal表示を行ないます。

10次に4をキーインすると図5のように帯グラフを描きますが、グラフのパターン分けは5つまで行ない、それ以上は分割線だけとします。

115をキーインすると図6のようにバレート図を描き、スクリーン・メッシュが入れられます。

12以上でグラフ機能の説明を終ります。次に、オプション・パッケージを呼び出すためにF10をキーインして命令待ちに戻り、F7をキーインしましょう。すると、“小計計算…1、集計…2、テーブル検索…3、自由計算…4”と表示します。

131をキーインすると“ソートシタ コウモク ヲ シテイシマスカ?”と聞いてきます。この意味はもし項目を指定

すると、その項目の同じものを1つのグループとして小計を行なうということ、たとえばランダムに入れた売上データを顧客名でソートして（CMT-3）おけば、自動的に顧客ごとの小計を出してくれま。

14Nをキーインすると“ハンイ ハ?”と聞いてくるので希望の数値を入れると指定の行番号までの小計を次々に出し、最後に全体の合計を出します。もちろん、印刷も指定により行ないます。

15次に2をキーインすると集計を行ないます。この機能は第1項目を見出しとして第2項目に集計値を入れます（この場合、第1項目と第2項目の桁数の合計を12桁以下になるようにしてください）。したがってあらかじめ第2項目に“0”を入れておけば選挙の得票集計のように使えます。部品や材料の集計表を作るときなどに便利です。行番号をキーインすると第2項目がその部数1つずつ増加します（カウンタとして使う場合）。行番号±数値をキーインするとその数値だけ増減します。材料集計や積算のとき便利です。

163をキーインするとテーブル検索となります。これには通常のテーブル検索とケーブルや安全スイッチのサイジングをするときのように安全最小値を検索するものの2つがあります。

17通常のテーブル検索を行なう場合は、たとえば値段表をサブファイルに入れておくと、メイン・ファイルの第1項目（たとえば品名）をキーとしてサブファイルの第1項目に取り込むので、ランダムに入れた品物の値段をたどころにファイルに記入してくれます。なお、サブファイルに該当がない場合は“?”を入れるので後で修正しましょう。

これはコスト・テーブルや材料テーブルからの数値の取り込みに便利です。2をキーインしたときの集計機能と組み合わせると集計→コスト計算までがスムーズにできます。

18サイジング・テーブル検索は、テーブルのサイジングを行なうときのように第1項目をデータとしてサブファイルの第1項目を検索しつつ比較し、安全側の最小値をサブファイルの第2項目からメイン・ファイルの第3項目に取り込むので、ずばりサイジングを行なうときに使います。

194をキーインすると自由計算となります。AUTO-2001は各項目間での演算をすべての行について行なうことを基本にしていますが、ときどき、任意の位置の個々のデータ間の計算を行ないその結果を任意に書き込みたいという要望があります。そのためのものですから自由に活用してください。

使い方は、行と列を指定するとそのデータがA=100、

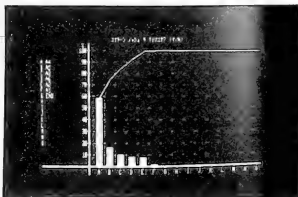
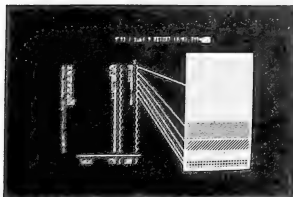


図 1

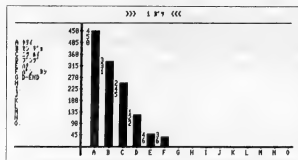


図 3

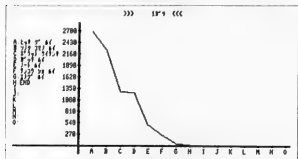
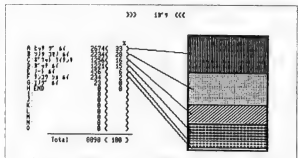


図 5



B=25, C=-10というふうにABC…に取り込まれます。あとはこのABC…間で演算をさせると結果を表示して「ドコニ イレマスカ?」と聞いてくるので、任意の行と列を指定するとそのデータを消して演算結果をそこにしてくれます。行、列の指定方法は25:Bまたは4:Cのように行番号:項目記号という形でキーインします。

キャラクタ・グラフィック同時 コピープログラムのリロケート

I/O'82年4月号のプログラム\$E000~\$E0CFFを\$F000~\$FCFFに転送し、転送したプログラムのうち\$F000~\$F26Aの間の3バイト命令のアドレスを\$E×××から\$F×××に変えればOKです。このようにしてできたプログラムをAUTO-2000 CMT-2の直後に、

```
FILE NAME: COPY $F000
S-ADR. $F000
E-ADR. $FCFF
J-ADR. $1300
```

でSAVEしてください。

なお、この藤原氏のプログラムはMZ-80BP5型プリンタ用なので、もしP6型を使用される場合は以下の変更を

図 2

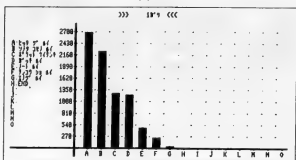


図 4

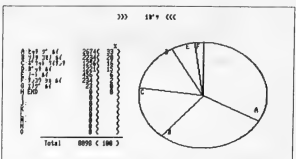
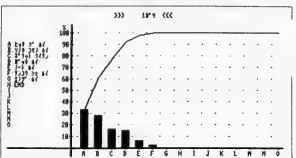


図 6



行なってください。

- ①「COPY \$F000」の\$F173…3E0F→3E00
- ②「AUTO-2001 CMT-2」の行番号2735のIFSG\$="Y" THEN USR(\$F000)→IFSG\$="Y" THEN PRINT/P CHR\$(20):USR(\$F000)にしてください (この方法はI/O'82年2月号の井上氏に教えていただきました)。

おわりに

AUTO-2001 CMT-2ではグラフィックと特殊な機能を集めたものになりましたが、このような機能は使用目的が一致していないとその効果は出ません。多分、皆さんが使う場合は大巾な手直し、追加を必要とすることでしょう。しかし、AUTO-2001の生命はこのようなユーザーの好みに合わせて自由に変更できることにあると思います。

今回のプログラムではAUTO-2001のデータを自由に出し入れるための機能が中心になっていますから、皆さんのアイデアでこれを生きたものに改造してください。また、グラフィックも「このようなことができますよ」という紹介にとどまったような気がします。グラフの多重表示、グラフィックのスクロールによる表示データの制限解除など、心残りの点が多いのですがご容赦ください。

AUTO-2001 TOSの設計について

先にご紹介したAUTO-2001はいかがでしたか。MZ-80Bの特長を生かしたTOS（テープ・オペレーティング・システム）を附加したことである程度満足していただいていることと思いますが、今回は皆さんが自分のソフトにTOSを組み込まれる参考に、MZ-80BにおけるTOSの考え方について述べたいと思います（説明の都合上AUTO-2001のプログラムをベースにして話を進めます）。

TOSの考え方

最初に少し余談になりますがOS（オペレーティング・システム）とはデータ処理装置の資源を制御する手続きで、計算機操作において人間の行なう仕事をできるだけ計算機自体に行なわれることにより効率を高めようというシステムです。パーソナル・コンピュータにおけるこのようなOSは記憶媒体で分類すれば、

OS — DOS ディスク
TOS テープ(今回のプログラム)

機能上の分類としては、

OS — リアルタイム処理 — タスク管理
データ・ファイル管理
バッチ処理 — ジョブ管理
データ・ファイル管理(今回のプログラム)

が考えられます

今回は、このデータ・ファイル管理のうちAUTO-2001用のデータ・ファイルのみの管理を行なうものを対象にします。

ではデータ・ファイル管理とは何かということになると、一般的にはコンピュータの機能を利用したテープ自動管理システムの構築ということになり

データ・ファイル管理 — データ・ファイル情報の管理
テープ操作の管理(自動化)

ということになるでしょう。

また、テープのデータ・ファイル情報の管理ということはAUTO-2001用のすべてのテープのデータ・ファイル进行管理するという立場と、1本のテープの個々のデータ・ファイル情報を管理するという立場があります。今回は1本のテープ管理について述べるので、TOSとはいっても非常に限定されたものであることをお断りしたいと思います。

TOS設計の前準備

①テープの種類と限定

本格的に作るのであればテープの種類への対応も考えられますが、実用的にはテープの種類を限定してプログラムの簡素化を計った方が得策です。ここではシャープの標準C-15ということにします。

②テープ・コントロール特性の実測

設計の前にMZ-80BにおけるC-15の諸データを測定して置く必要があります。必要なデータは次表のようにな

りますが、C-15以外のテープで設計する場合は再測定してください。

テープの長さ	178カウント	カウンタの読み
早速りのスピード	6.3カウント/秒	ただし立ち上り1秒の遅れ
1レコード(255バイト)の長さ	3.0カウント	テープ全体の最大値

どうも実際に使ってみると、C-15ではやや記録容量が小さく、C-30またはC-45あたりが「実用的にはよいようです」。

③カセット・コントロール・サブルーチン

ご存知のようにMZ-80Bはオート・カセット・コントロール機能を持っていますが、その制御は8255によって行なわれており、入出力ポートとコントロールの内容は「SHARP OWNER'S MANUAL」を見ていただくことにして必要な機能をサブルーチンとしてBASICで記述してみます（番号はAUTO-2001の行番号と考慮してください）。

モーターストップ	6310 OUT#227,13:OUT#224,24 :OUT#224,18:RETURN
巻戻し	6600 REW#16 6600 OUT#224,16:OUT#227,11:OUT#227,10 :OUT#224,17:OUT#224,18:RETURN
早速り	6305 FAST#16 6305 OUT#224,18:OUT#227,11:OUT#227,10 :OUT#224,19:OUT#224,18:RETURN
無音検出	6405 J=100 6410 INP#225,X:IFX THEN6405 6415 J=J-1:IFJ THEN6410 6420 RETURN

ここで少し説明を加えると、SB-5520はビット処理がいくらか面倒なので、ポートAのビットのうち、PA4（表示画面全体のB/Wを反転させる）は「H」、それ以外は「L」となるようにしてノーマル状態を設定し、単純化しているので注意してください。

無音検出はJの値を変えることにより無音の確認時間を変えることになりました。すなわちREW完了の確認はJ=100程度、レコード頭出しのための確認はJ=5程度が良いでしょう。

TOS仕様の決定

では、いよいよ仕様を定めて設計に入ることにしましょう。すでに述べたように、今回のTOSは1本のカセット・テープに入っているAUTO-2001用のデータ・ファイル管理を行なうものですから、目標としてはシャープ標準のディスク・ベースの持つファイル管理の利便性が手頃と思われる。これに相当するものとして、

1500 GOSUB7100:CURSOR0,0:PRINT" テープ" II テープ"ニ セーブ"---
アラク テープ"ヨリ ロード"--- 巻 ファイル リスト--- 巻 ファイル ワク"ヨリ"---

を設計することにしよう。

①「ファイルリスト」はテープに記録されているデータ・ファイル情報を表示する機能とします。ディスク・ベースのDIRです。

②「テープニセーブ」はメモリ上にあるコントロール下のデータ・ファイルをテープにセーブする機能とします。ディスク・ベースのSAVEです。同じファイル名を指定した場合は書き替えるかどうかを確認してくることにし

ましょう。

①「アタニテープヨリロード」はテープのファイル・データをロードする機能としますが、データ・ファイルの選択はファイル・リストのファイルNo.で行なうことにしましょう。ディスク・ベースのLOADです。

②「ファイルサクジョ」はデータ・テープの任意のデータ・ファイルを削除する命令です。ディスク・ベースのDELETEです。

③以上の機能をすべて自動で実現し、さらにデータ・ファイル管理情報をテープとメモリ間で受け渡し機能(MOUNTとDISMOUNT)は自動的にこなすことにしましょう。

④その他の機能としてディスク・ベースではLOCK, UNLOCK, RENAME, COPY, などがありますが、今回は別愛します(後で述べますがこれらへの対応は他の方法で可能です)。

また、欲をいえばテープ管理用のVOL.No.を登録したい所ですが、このような機能の追加は皆さんで行なってください。ただしテープのイニシャライズ(新規テープの場合)は必要となりそうです。

TOSの設計

①ファイル構成

データ・テープの頭にデータ・ファイルの情報としてのファイル・リストと実際のデータが記録されている位置をボスアドレス・マップを記録し、データ・テープをセットしたとき、最初にメモリ上に読み込む方式とします。

このファイル・リストとアドレス・マップは具体的にはC-15テープの最初から16カウントまでを記録の領域とし、16カウントから178カウントまでを28に分割してデータ・ファイルの記録領域とします。

この28に分割されたものに1番から28番までの7番地を付けて、記録されている番地にデータ・ファイルNo.を書き込んだものがアドレス・マップで、記録のない番地には「0」を記入することにします。したがってテープ・イニシャライズのときは全番地を「0」にして置くことになります。

(C-15テープへの記録状態)

0カウント		16カウント					178カウント			
ファイル・リスト+アドレス・マップ		1番地		2番地		27番地		28番地		
アドレス・マップ										
アドレス	1番地	2	3	4	5	6	26	27	28	
ファイルNo.	1	1	2	2	0	3	0	0	0	

(ファイル・リスト)

No.	ファイル名	コメント	シフト	リット
1	CONSOLE	CONSOLE	57	1/28
2	INSTR 1	インストラクション	57	1/28
3	INSTR 2	インストラクション	57	1/28
4	INSTR 3	インストラクション	57	1/28
5	INSTR 4	インストラクション	57	1/28
6	INSTR 5	インストラクション	57	1/28
7	INSTR 6	インストラクション	57	1/28
8	INSTR 7	インストラクション	57	1/28
9	INSTR 8	インストラクション	57	1/28
10	INSTR 9	インストラクション	57	1/28
11	INSTR 10	インストラクション	57	1/28
12	INSTR 11	インストラクション	57	1/28
13	INSTR 12	インストラクション	57	1/28
14	INSTR 13	インストラクション	57	1/28
15	INSTR 14	インストラクション	57	1/28
16	INSTR 15	インストラクション	57	1/28
17	INSTR 16	インストラクション	57	1/28
18	INSTR 17	インストラクション	57	1/28
19	INSTR 18	インストラクション	57	1/28
20	INSTR 19	インストラクション	57	1/28
21	INSTR 20	インストラクション	57	1/28
22	INSTR 21	インストラクション	57	1/28
23	INSTR 22	インストラクション	57	1/28
24	INSTR 23	インストラクション	57	1/28
25	INSTR 24	インストラクション	57	1/28
26	INSTR 25	インストラクション	57	1/28
27	INSTR 26	インストラクション	57	1/28
28	INSTR 27	インストラクション	57	1/28

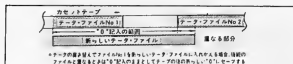
- リスト エント -

ファイル・リストには上表の情報の他にたとえばデータ・ファイルの項目数やデータ数も欲しいところですが、そのときは項目数はIN(F)、データ数はDE(F)ですからそれをファイル・リストに書き込むようにすればOKです。

②データ・セーブ

セーブすべきデータのサイズから必要なテープ長を算出しそれを必要な番地数に直して、データ・マップの最初から「0」が記入されている番地を探し、その「0」がセーブに必要な番地数だけ続いているなければさらに次の「0」を探し

て必要な番地数だけ続いている場合はそこにセーブすると共にファイル・リストの更新とデータ・マップの記入を行なう。



③データ・ロード

要求されたデータ・ファイルNo.がどこに記入してあるかをアドレス・マップで調べその頭にジャンプしてロードを行なう。

④ファイル・サクジョ

要求されたデータ・ファイルNo.をアドレス・マップで調べその番地を「0」に書き替えると共にファイル・リストを書き替える(ついでにファイルNo.をつめることにしましょう)。

⑤番地へのジャンプ

これがTOSの中心となるわけですが、簡素化のため内部時計によるFAST走行時間制御としましょう。

高級な方式としてはロード、セーブで止まった位置から新しい位置へ相対的にジャンプするようにして時間短縮する。また正確に頭出しを行なうために無音部検出(データ書き込みの頭に5カウントの無音部がある)を行なって位置決めをするなどがありますが、これは皆さんで工夫してください。

ここではデータ・テープは常に0カウントの位置に巻き戻されており、定められた番地へのジャンプは内部時計にタイマーをセットして(FOR NEXTのタイマーでは正確になります)FAST時間を定めるオープン・ループ・コントロールにしています。

いままでのところ、誤差±1カウントで充分実用に耐えていますからご安心ください。

⑥巻き戻し

テープを巻き戻して常に初期状態を保つためREWを使っていますが、この命令は完了の検知を行いません。したがってREW完了を確認するため無音部検出方式を採用しました(多分もっとよい方法があるでしょう)。この考え方はREWが完了したらテープの読み取りヘッドからの信号がなくなるので無音状態が約3秒続いたらREW完了とみなすという方式です。

⑦データ書き直し

データ・セーブのところでファイル名が同じ場合は「オナジナメス / ファイル ガ スデアリマス」という確認をして書き替えるの指示が出た。まず①のサクジョを行ない②のセーブに入る方式とします(前図参照)。

TOSのプログラム化

以上で基本的な考え方の説明を終ります。

プログラム化は皆さんにとって蛇足になると思いますが、AUTO-2001のプログラムで説明しましょう。

①ファイル・リスト

行番号6000-6006がファイル・リストとアドレス・マップの読み込みで、6100-6115がファイル・リストの表示です。

②データ・ロード

行番号6200-6202が指定されたファイルNo.のアドレス

・マップの番地の検索で、6300～6310が番地へのジャンプです。D=3+Xとすることにより16カウントからが1番地になります。

また、C-15テープの場合FASTによる1秒当りのテープ走行長を基準に番地分けをするとうちょうど具合がよかったので、このプログラムではそのようにセットしています。すなわち1秒走行長が番地割り付けのテープ長というわけです。

また行番号6305はFASTで書き直してもOKです。

⑥データ・セーブ

行番号6210～6236がセーブできる空き番地を探するルーチンです。6210のAはAUT0-2001データのサイズによって必要な番地数を算出して用いるもので、3カウント/レコードと6.3カウント/番地の関係を使っています。また、6116～6219はこのAがどこの番地からなら入るかを調べている部分で、入らない場合は6214でテープの入れ替えを要求します。

⑦ファイル・サクジョ

行番号6230～6236がファイルのサクジョです。ついでながら先月の説明ではファイル・サクジョのときファイル・リストの自動更新を行なわないで、DISMOUNTのときC L命令で、ファイル・リストの書き替えを行なうようにしていましたが、これはどうやら自動更新の方が間違いないようです。そのための変更は、行番号1554を、

```
1554 GOSUB6230:CURSOR0,24:PRINT"## シフト オマセ クサライ ##"
      GOTO1660
```

に書き直してください。

おわりに

MZ-80Bはその高速度性とテープ・コントロールさらには64Kバイトのクリーンさからマニア向きの非常にコスト・パフォーマンスの高い機種だと思います。

その機能を引き出す1つの手段として、今回、BASIC記述のTOSを紹介したわけですが、当然これらの仕様は皆さんらSB-5520自体に機械語でビルトインできるもの

です。是非挑戦し、発表していただきたいと思います。どのようなテープにも対応でき、さらにはテープ・システム全体の管理まで可能なTOSを内蔵したSB-5520のニューバージョンが遠からずI/O誌上に登場することを期して筆を置きます。

追記

TOSの仕様決定の⑥でLOCK,UNLOCK,RENAMEへの対応を約束したので要点のみ述べてます。

* *

LOCK,UNLOCK機能がないために生ずる不都合は不用意なDELETEまたはイニシャライズによるものですが、この場合、データ記録そのものは生きていますので、次の操作で回復できます(データ・ファイル・テープのコピーも同じ要領で達成できます)。

①CMT-1をLOAD:RUNして前のデータを使うと選択するとファイル・リストを表示して“ファイルNo.”と聞いてきます。

②この場合ファイル・リストから消えているためファイルNo.が入れられないのですが、[RE]をキーインするとファイル名を聞いてくるので、テープを、元の記録であったカウントまで送ってファイル名をいれると、消えたデータ・ファイルをロードします。

③あとはこのファイルをセーブすれば完全に回復します。

* *

次にRENAMEですが以下の手順でできます。

①RENAMEするデータ・ファイルをロードします。

②このデータ・ファイルをセーブすると“オナジナエノファイルガリマス。カキカエマスカ(Y/N)?”と聞いてくるので[N]をキーインし新しい名前を入れてセーブします。

③前のファイルをサクジョします。

④以上の手順が面倒な方は、せこいやり方ですがBASICに抜けて?L\$(I)をやって任意のファイル・リスト情報を読み出し、新しく書き込んでGOTO1660とダイレクトでファイル・リストを書き直せばOKです。

AUTO-2001 CMT変数表

(1)作業用変数

A S, B S, C S, X S, Y S, Z S, C C S, F C S, F D S (26), U S (26), A, B, C, D, E, H, I, J, K, L, M, N, Q, W, Y, W (26), X (26), Y (26), C I, C 2, C 3

役割り

I, J=FOR NEXT用 Y=カーソル・コントロール Y S=書き替えコントロール
F D(I)=プリント・コントロール F D S (I4)=自動プログラムのJUMP
F C S=ファイル名 C I-C 3=最初のロード予定、命令コントロール

(2)指定変数

F, G, D S (1,255), I S (1,26), S S (1,26) I C (1,26), I P (1,26), I N (I), I E (I), D E (I), F T S (I), T S (26), T (I), L S (I), K S (I), F I S (I), F 2 S (I), F 3 S (I), F 4 S (I), F 5 S, F 6 S (I),

役割り (Fファイルで1行J項目とした場合)

F=コントロール下ファイル G=書ファイル
D S (F, I)=ファイル・データ I S (F, J)=項目名
S S (F, J)=項目の属性 I C (F, J)=項目の所番
I P (F, J)=項目の位置 I N (F)=項目数
I E (F)=項目の読込所番 D E (F)=データ数
F T S (F)=ファイル名,コメント,日付 T S (F, I)=1番目の自動プログラム

T(F)=自動プログラム式数

式

K S (I)=TOSのアドレス・マップ L I (I)=ファイル・リストの内容

* F I S (F)=F 6 S (F)は項目およびデータ表のFORMAT

F I S (F)=	F 2 S (F)=	名前	住所
F 3 S (F)=	F 4 S (F)=		
F 5 S (F)=	F 6 S (F)=		

(3)使用例

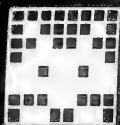
1行J列のデータはM I D S (D S (F, I), I P (F, J), I C (F, J)-I), これをとえばX Sと入れ換えると新しいD S (F, I)=L E F T S (D S (F, I), I P (F, J)-I)+X S+T+R I G H T S (D S (F, I), I E (F)-I P (F, J)-I C (F, J)+I)

(4)主なサブルーチン

5000=警告音	5050=A B C入り項目表
5070=A B C, a b c入り項目表	5100=タイトル, 日付表示
5150=	5200=ワード・プロセッサ用1行入力
5210=ワード・プロセッサ用1行読み取り	5250=行番号指定
5300=エラー警告	5400=項目設定
5500=項目文字数管理	5600=項目文字のE D I T
5700=項目表示表	5900=データ1行E D I T
5980=メイン・サブ表示	6000=ファイル・リスト読み込み
6100=ファイル・リスト表示	6210=TOS空番地サーチ

200

【作者】 王向华、王向华、王向华、王向华



TV 黒板



(5) PEN UP
PEN DOWN

(8) WRITE
ERASE

(6) 4 5 6
2 3 4



MZ-80B

■高嶋 晃

今回紹介するのは、MZ-80B用TV黒板プログラムです。
'81年11月号のグラフィック花札ゲームの華麗な図柄に目を
みはった方も多かったのではないのでしょうか？あの10Kバ
イトにもおよぶグラフィック・データを作成したのがこの
プログラムです。

元来は小生のグラフィック・ゲーム開発用に作成したも
のであり、門外不出の秘密プログラムの予定だったのですが、
I/O編集部から「プログラムの出し惜しみは体に良くない」と
いう要望がありましたので発表することにしました。

このプログラムは約2.5Kバイトの小さなものですが、な
かなか便利で重宝しています。グラフィック・データ作り
には非役立ててください。

概要

もしかしたら、グラフィック・データを作るのに、文房
具屋さんから方眼紙を買ってきて、マス目を図柄に沿って
塗りつぶし、それを一生懸命計算しながら数値データに変
換している人がいるのではないですか？ドキッノとしたあ
なた、そう、あなたのことですよ！

時代の最先端をいくパソコンのオーナーであるあなたが
そんな原始的手法を取っているということは取すべきこと
なのです（かく言う小生もそうでしたが…）。

本プログラムは、その原始的手法でやっていたことをパ
ソコンにやらせようというものです。

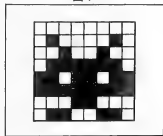
使用方法

具体例に沿って説明した方が解り易いと思いますので、
図1の図柄をデータ化してみることにします。

①RUNすると、最初に「文字変数にデータ・インプット…
1」「メモリにデータ・インプット…2」のメニューが表示
されます。いま、仮りにメニュー1を選択したとして話を
進めます。

次に「データ名は？」と聞いてきますので、後日、何の
データであったのか解り易いように適当な名前をつけてお
きます。この場合は「インバーダ」とでもしておきましょ

図1



例1

```
60010 REM インバ-ダ- X=B Y=B  
CHR$(0)+"B$~0n"+CHR$(24)+"f"
```

例2

```
60010 REM インバ-ダ- X=B Y=B  
バ-ンチ = 60000~60007
```

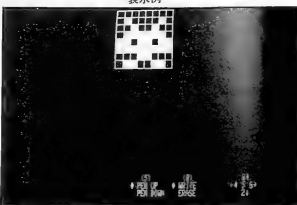
サンプル1

```
FOR I=0 TO 7:A$=A$+CHR$(PEEK(60000+I)):NEXT
```

う。
その次は図柄の縦横のドット数を聞いてきます。「縦=」
に対して、この場合は8を、「横=」に対して8を入力する
と、画面に8×8の枠が表示され、ドット・カーソルが右
上に点滅します。

②あとは、テン・キーを使用して好きな絵を書いてくださ

表示例



い。

[2], [4], [6], [8]の各キーは、カーソルの上下左右の移動用です。[5]のキーはペンのUP, DOWNに、[0]のキーはドットをセットするか、リセットするかの切り換えに使用します。つまり、[5]のキーを押してペンDOWNにし[0]のキーを押してWRITEした後、カーソル移動すれば線が描かれていきます。

書き終われば、[RETURN] キーを押します。

⑥画面には例1のように表示された後、プログラムはSTOPするので、カーソル移動して60010の行に持って行き、[RETURN]キーを押します。同様にしてカーソルを 60020の数字の次に持っていき、文字変数を書いた後、[RETURN]キーを押せば、グラフィック・データのREMARKが60010行に、データが60020行に作られたこととなります。

グラフィックを数値に変換した際、ASCIIコード29以下はコントロール・コードなので画面に表示できないため、やむを得ずCHR\$(X)で書かれますが、それ以外はメモリ節約のために“文字”で表わしています。

しかし、図柄を書けばわかりますが、ASCIIコードが29以下のデータが室外多いものでして、化札ゲームのように膨大な量のデータですとメニュー1でデータを作れば、すぐにメモリ・オーバーしてしまいます。

そんなときにメニュー2を使います。メニュー2はデータの数値を直接メモリに書いていくもので、どんなデータも横1列8ビット分のデータは1バイトの消費でおさまります。REMARKは例2のように表示されます。

メニュー2を使用するときは、RUNする前にLIMITをかけてマシン語領域を確保しておくをお忘れなく。

なお、メニュー2の長所はメモリ節約ですが、欠点はデータの修正、判読がにくいという点です。従って、メモリに余裕のある場合はメニュー1をお勧めします。

メニュー2を選択した場合のデータを文字変数に収納する方法をサンプル1に示します。このサンプルでは60000

～60007番地の内容がA\$に収納されています。

⑦グラフィック・データをすべて作り終わったら50000行から55900行を消去し（DELETEコマンドのないのが残念！）、メイン・プログラムを作成してください。くれぐれも、この順番を逆にしないようにしてください。逆になるとMERGEコマンドがないために、“TVコクパン”をキーインしなければならなくなります。

⑧なお、ご使用に当たっては“使用上の注意”をよくお読みになった上、服用してください。万一、幻覚症状、手指の痛みなどの症状に気付かれた際は、ご使用を避け、早く就寝されるように願います(!?).

使用上の注意

①縦横ドット数を決める際、シャープBASICの制約上、横方向は8ドットの倍数とし、縦×横のドット数が $256 \times 8 = 2048$ を越えないようにしてください（BASIC内部でのストリング・データの最大文字数は255文字のため）。

②シャープBASICの制約上、PATTERN文の縦方向積み重ね段数は24ドットまでなので、それ以上の積み重ね段数の場合は縦方向を24ドット単位に分割してデータを作ります（自動分割されますが、表示の際に、このことを考慮してください）。

③ドット・カーソルを点滅させているためたとえばWRITEの場合ドットをA地点から、隣の地点まで移したとき、移した時点でA地点のドットが登録されます。従って、最後の点を打ち終わったら、ひとつ前の点の位置へカーソル移動してから[RETURN]キーを押してください。

④いままでの説明がわかりにくかった方へ、MZ-80BのBASICのマニュアルのPATTERN文の説明をもう一度よく読んだ後、本稿を読み返してください。

TV黒板 プログラム・リスト

```

50000 REM #####
50001 REM # TV コクパ Rev 1.0 #
50002 REM # 1981.12 #
50003 REM # PROGRAMED BY A.TAKASHIMA #
50004 REM #####
50010 CONSOLEC80
50015 GRAPH I1,01
50020 P=60000
50022 PRINT" TV コクパ メニュー"
50025 PRINT"もし'ヘルプ'ニ'テ'インプット....1":PRINT"メモリニ'データ'インプット....2":INPUT
ME
50027 IF ME>2 THEN50025
50030 INPUT"データ'メイ'ハ' ?":NA$
50035 IF ME=2 THEN INPUT"データ'カギ'コ'メ'ン'チ'ハ' ?":PP:PQ=PP
50040 PRINTCHR$(6):GRAPHIC:INPUT"データ"=YZ:INPUT"ヨコ(Bノ'ハ'イズ)"=XZ:CH=YZ*XZ/8
50045 X1=1:Y1=1:W=1:U=1:N=0
50050 DIM D(200)
50055 PRINTCHR$(6):GRAPHIC:LINE0,0,XZ+1,0,XZ+1,YZ+1,0,YZ+1,0,0
50060 CURSORB,22:PRINT"(5) (0) 8↑"
50065 CURSOR5,23:PRINT" ● PEN UP ● WRITE 64 5 6"
50070 CURSOR5,24:PRINT" ● PEN DOWN ● ERASE 24:"
50100 C=POINT(X1,Y1)
50110 I=POINT(X1,Y1):IF (I=1)+(I=3) THEN RESET X1,Y1:GOTO50120
50115 SET X1,Y1
50120 GOSUB51000
50125 IF (X=0)*(Y=0)*(N=0) THEN50110
50130 IF U=1 THEN GOSUB52000:GOTO50160
50140 IF W=1 THEN SET X1,Y1:GOTO50160
50150 RESET X1,Y1
50160 X1=X1+X:Y1=Y1+Y
50170 IF N=1 THEN55000
50180 GOTO50100

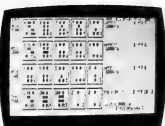
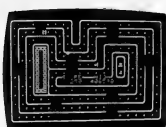
```

TV黒板 プログラム・リスト

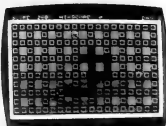
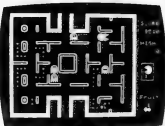
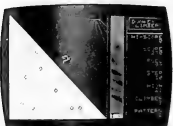
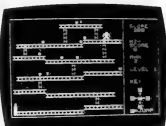
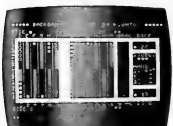
```

51000 X=0:Y=0:FOR I=0 TO 50
51005 GET G#:IF G#="" THEN51290
51007 ON ASC(G#)-47 GOTO51010,51020,51030,51040,51050,51060,51070,51080,51090,5
1100
51008 GOTO51280
51010 IF W=1 THEN W=0:CURSOR17,23:PRINT" ":CURSOR17,24:PRINT"◆":RETURN
51015 W=1:CURSOR17,23:PRINT"◆":CURSOR17,24:PRINT" ":RETURN
51020 X=-1:Y=1:GOTO51110
51030 Y=1:GOTO51110
51040 X=1:Y=1:GOTO51110
51050 X=-1:GOTO51110
51060 IF U=1 THEN U=0:CURSOR5,23:PRINT" ":CURSOR5,24:PRINT"◆":RETURN
51065 U=1:CURSOR5,23:PRINT"◆":CURSOR5,24:PRINT" ":RETURN
51070 X=1:GOTO51110
51080 X=-1:Y=-1:GOTO51110
51090 Y=-1:GOTO51110
51100 X=1:Y=-1
51110 IF (X+X1<1)+(X+X1>XZ) THEN X=0:Y=0
51120 IF (Y+Y1<1)+(Y+Y1>YZ) THEN X=0:Y=0
51130 RETURN
51280 IF ASC(G#)=13 THENN=1:RETURN
51290 NEXT
51300 RETURN
52000 IF (C=1)+(C=3) THEN SET X1,Y1:GOTO52020
52010 RESET X1,Y1
52020 RETURN
55000 REM
55010 YP=-1:Y1=INT(YZ/24):Y2=YZ-Y1*24
55020 YP=YP+1:YS=YP*24:YE=YS+24:IF YP=Y1 THEN YE=YS+Y2
55022 IF YS>YE THEN55900
55025 P1=0
55027 P=P+10:PRINT:PRINT:PRINTP;" REM ";NA$;" X=";XZ;" Y=";YE-YS;
55030 IF ME=2 THEN PRINT" \"/"&" ;PP;"^";
55035 FOR X=1 TO XZ STEP 8
55040 FOR Y=YS+1 TO YE
55050 J=256:D=0:FOR Z=0 TO 7
55060 J=J/2:I=POINT(X+Z,Y):IF (I=1)+(I=3) THEN D=D+J
55070 NEXT
55080 IF ME=2 THEN POKE PQ,D:PQ=PQ+1:GOTO55190
55090 IF P1<158 THEN55120
55100 IF F1=1 THEN PRINTCHR$(34);
55110 P1=0
55120 IF P1=0 THEN F1=0:F2=0:P=P+10:PRINT:PRINTP;" " ;P1=12
55130 IF (D>29)*(D<>34) THEN55160
55140 IF F1=1 THEN PRINTCHR$(34);;" ";F1=0:P1=P1+2
55145 IF F2=1 THEN PRINT" ";P1=P1+1
55150 F2=1:PRINT"CHR$(";STR$(D);")":P1=P1+9:GOTO55190
55160 IF F2=1 THEN PRINT" ";CHR$(34);:F2=0:P1=P1+2:GOTO55180
55170 IF F1=0 THEN PRINTCHR$(34);:P1=P1+1
55180 F1=1:PRINTCHR$(D);:P1=P1+1
55190 NEXT Y,X
55200 IF F1=1 THENPRINTCHR$(34);
55210 IF YP<Y1 THEN GOTO55020
55890 IF ME=2 THEN PRINTPQ-1:PP=PQ
55900 PRINT:PRINT"RUN50030":STOP
60000 REM ** DATA **

```

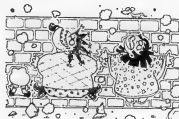


BLOCKくずし	ルリタテハ	208
SPACE INVADERS	FALCON. S	213
HEAD-ON PART 2	ルリタテハ	220
グラフィック花札こいこい	高嶋 晃	227
4人ポーカー	大場啓二	240
チェス・プログラム	馬場隆信/COMPAC	247
BACK GAMMON	大藤謙二	271
CRAZY CLIMBER	ばっくまん	280
DONKEY KONG モドキ	居候 VALZA	287
DONKEY CLIMBER	ばっくまん	297
PUCK MAN	ルリタテハ	306
DAN GAME	ばっくまん	313



MZ-80B(マシン語)

01000



くずし



ブロックくずしはもうすでに過去の遺物となってしまいましたが、古典的名作と言えると思います。これが自分の機種にないというのはちょっと寂しいので作ってみました。原型のままで見つからないので、動く反射器を付けたりして少しは工夫してあります。久しぶりにやってみるとなかなか面白くて熱中してしまいました。

注) このゲームではグラフィックRAM MZ-8BGが必要です。

■ルリタテハ



ゲームのやり方

ゲームの内容はどなたもご存知と思います。プログラムはオール・マシン語ですから、8 A 6 A 番地からスタートさせます。まず、タイトルが出てくるので、**[CR]** (または**[ENT]**) キーを押してください。するとブロックが現われ、上の反射器が左右に移動します。下の反射器は固定されています。

ラケットの左右移動はテンキーの **[4]** と **[6]** です。サープはテンキーの **[5]** です。 **[5]** キーを押すと、BALLがブロックのすぐ下のところから出てきます。あとはひたすらラケットで打ち返してブロックを全部消してください。

BALLのスピードは遅く感じるかもしれませんが上の壁にぶつかるスピードは倍になるので安心してください。

ブロックの得点は画面左側に表示されているように、一番下の行が10点で、上に行くにつれて10点ずつ増えていきます。1面が終了すると終了の音楽が流れて第2面になります。ブロックの得点がそれぞれ10点ずつアップします。

ブロックはBALLが上下から当たったときのみに消え、左右や角から当たったときはBALLは弾ね返ってしまいます。

ハイ・スコアはプログラム中のワーク・エリアにあります。そのため、ゲームをしてからテープにSAVEすると、そのハイ・スコアは次にLOADしたときにも保存されています。ハイ・スコアを0にするには、84 E 3 番地と84 E 4 番地に00を入れてください。



プログラムの入力

オール・マシン語のプログラムですから、モニタを使って打ち込みます。BASICから**MON [CR]** とやるとモニタ・モードになります。次に**M [CR]** とすると画面に

M-ADR. \$

と出ますから8000と打ち込んで**[CR]** してください。あとはプログラムのリストどおりに16進数を打ち込んでいきます。80Bのモニタでは1つ打ち込むごとに**[CR]** を押さないといけないので面倒くさいのですが、まあ仕方ないでしょう。

全部打ち込んだら、RUNする前にテープへSAVEしておきます。Mコマンドは**[BREAK]** キーを押すとコマンド待ちに戻りますから、テープをセットして**S [CR]** とすると、

FILE NAME :

と出ますから好きなNAMEを付けてください。

次に、S-ADR. \$に8000、E-ADR. \$に8AE0、J-ADR \$に8 A 6 Aと打ち込んでSAVEすればオート・スタートテープになり、次にLOADしたときには自動的に始まりです。SAVEをしたら必ずベリファイをしておきましょう。

RUNさせるのはまだ早いですよ。次に、CHECK SUMを調べます。プログラム内にワーク・エリアがあるので一度動かすとあちこちの値が変わってしまいます。CHECK SUMがすべてOKでしたら、もう一度テープに入れ直しておきましょう。

やっとRUNできます。コマンド待ちの状態から**J [CR]**

写真1 [CR]を押すと、ゲームが始まる。



写真5 壁にうまく穴をあけるのが、高得点のコツ

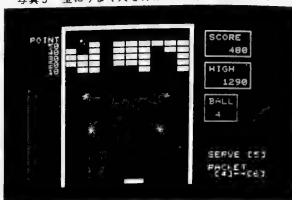


写真2 [5]キーを押すと、ボールがサーブされる。

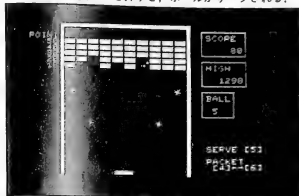


写真6 いったん上に入ると、ボールが速くなる。

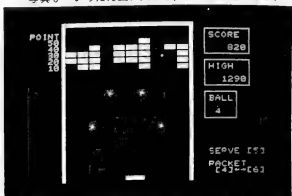


写真3 テンキーの[4]と[6]でラケットを操作するのはかなり難しい。



写真7 ブロックが残り少なくなると、消すのが大変!

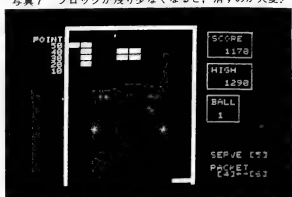


写真4 上の反射器はたえず左右に動いている(写真2~4)。

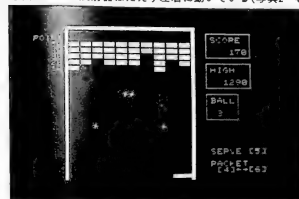


写真8 5個のボールを使い果たし、ゲーム・オーバー

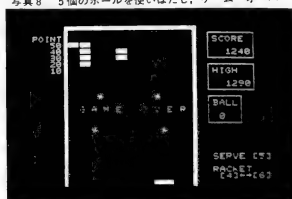
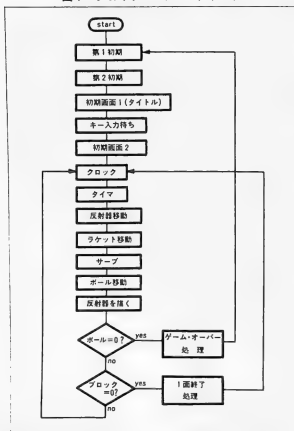


図1 ジェネラル・フローチャート



とすると、J-ADR. Sと出るので8A6Aと打ち込んで、**CR** するとゲームがスタートします。

モニタについての詳しいことは付属のモニタ・リファレンス・マニュアルを読んでください。



プログラムについて

MZはクリーン・コンピュータなので、しょっちゅうBASICがバージョン・アップされてしまうこともあるためBASIC部分を作ると後で困るのでは…と思います。オール・マシン語でプログラムしてみました。モニタ内のサブルーチンはMUSICの部分だけ使っているの、将来モニタの内容が変更になってもその部分だけを変えれば済みます。

プログラムはハンド・アセンブルで作ったためNOPだらけで、しかもハチャメチャな内容になっています。何しろ一度入力した部分はまた入れ直してという面倒なことをしたくないので、かなり無理をしているところもあります。

フローチャートなどという恐ろしいものはとても書けそうありませんが、一応ジェネラル・フローチャートだけを図1に示します。

各サブルーチンに付属するワーク・エリアやデータ・エリアはそのすぐそばに置くというひどい方法を使っているの、エリアはあちこちに点在しています。ワーク・エリアを表1に、データ・エリアを表2に示します。

もし逆アセンブルする方はこの2つのエリアをはずしてやるようにしないとんでもないことになります。各サブ

表1 ワーク・エリア

アドレス	内容
8082, 8083	乱数値
808B	メイン・クロック値
808D, 808E	ラケット・クロック値
81A9, 81AA	反射器1アドレス
81AB	方向
81AC~81AE	反射器2で上と同様
81B6, 81B7	ラケット・アドレス
84D5, 84D6	1行目のブロックの得点
84D7, 84D8	2行目のブロックの得点
84D9, 84DA	3 " "
84DB, 84DC	4 " "
84DD, 84DE	5 " "
84DF, 84E0	スコア
84E3, 84E4	ハイ・スコア
84E7	ボールの数
86A1	ボール移動のフラグ群 (BIT0は左右, BIT1は上下, BIT2はボールの型, BIT3はボールの出現を示す)
86A2	ボールの速度
86A3, 86A4	ボールのY値, X値
86A5, 86A6	ボール・アドレス
86AA	ブロックの数

表2 データ・エリア

アドレス	内容
819F~81A8	反射器のデータ
825E~8388	タイトル文字のデータ
8446~84D4	V-RAM (キャラクタ) に描く文字のデータ
861C~8658	MUSICデータ
86C4~86D2	ボール・データ
8A5D~8A69	ゲーム・オーバーの文字

表3 マシン語サブルーチン

アドレス	内容
8000~	10進出力サブルーチン
804D	タイマ
805F	乱数を発生
808F	クロックの更新
80A8	各種プリント・サブルーチン
80DF	壁を描く
8116	ブロックを描く
8175	反射器を描く
81BA	ラケットを描く
8201	タイトル文字を描く
8389	画面クリア
83BB	キャラクタ画面の文字を描く
84E9	初期画面1
8504	" 2
851E	反射器の移動
8577	ラケット移動
85EC	MUSICサブルーチン
8659	サーブ
86E5	得点計算と表示
8738	ボール移動のメイン・ルーチン
8777	ボール右上へ移動
880E	" 左上へ "
8881	" 左下へ "
892E	" 右下へ "
89BA	固定反射器を描く
89D0	1面終了処理
8A09	ゲーム・オーバー処理
8A6A	メイン・ルーチン
8AB0	初期設定1
8AD2	" 2

ルーチンの大まかな内容を表3に示します。

メイン・ループの中に **BREAK** キーを押したかの判定

を付けているので、ゲームの途中で **BREAK** キーを押すとモニタに戻ります。

80Bでマシン語プログラムを組むとき、便えそうなサブルーチンを2つ紹介します。

1. V-RAMエリア・オープン

80Bは64Kバイトすべてプログラム用のRAMなので、ただV-RAM(ビデオRAM)のアドレスに書き込んでも両面に表示されません。そのためには各種ポートのスイッチを入れることで、RAM後半をV-RAMにしてやらなくてはなりません。そのためのプログラムをリスト1に示します。

まず、PIOの第7ビットを1にしてV-RAMをオープンにし、次にOUTポートF4Hに02を出力することでグラフィック1のINPUT, OUTPUTを可能にしています。これに関してはオーナーズ・マニュアルのp.47付近を見てください。

BASICではPRINT文のとき、このようにV-RAMをオープンにして書き込んでから再びクローズしているみたいですが、この程度のゲームではメモリはたっぷりと余りますから、後半のV-RAMはオープンしたままで、いちいちクローズしていません。

このプログラムはリロケータブルですからこのアドレスにも置けます。

2. KEY IN

モニタのGETKYのサブルーチンを使ってもキー入力是可以しますが、これではリアルタイムなゲームには向きません。直接キーボードからの入力を見る方法はリスト2のようになります。

オーナーズ・マニュアルのp.54の表を見てください。Cレジスタに「キーストロープ値+10H」の値を入れてこのサブルーチンをコールすると、Aレジスタに結果が入って戻ります。押されたKEYに対応するビットが0になっていますから、ビット命令を使って調べれば、どのKEYが押されたかがわかります。

たとえば、Cレジスタに12H(キーストロープ値は2)を入れてコールします。このときはテンキーの0~7までを調べることができ、**4** キーが押されていたとすればAレジスタに入っている値のビット4が0になっています。このプログラムもリロケータブルです。

80Bでマシン語プログラムを作るのはこれが初めてで、最初はわからないことだらけで苦労しました。グラフィックを使っているので、これの使い方もなかなか大変です。ドットは1ドットずつ動かすわけではなく、4ドット分ずつ動くようにしてあります。ボールの形は2種類をあらかじめデータとしてストアしておき、それを使って描いています。

点数や文字はキャラクタV-RAMに書き込んでいます。2つのページを重ねられるというのはなかなか便利なもの

で、これはいろんなゲームに応用できそうです。



変更点

- 1) 全体のスピードは804E番地の値を変えます。大きくすると遅くなります。
- 2) ラケットのスピードは8098番地の値を変えます。01以下にはできません。ラケットのスピードは速過ぎるとちょっとキーを押しただけでも動き過ぎて制御できませんし、遅いとボールに追いつかないということになりなかなか難しいところです。
- 3) ボールの数は現在6個ですが、増やしたければ、8AB1番地の値を大きくします。ただし、10個以上ではほとんど表示がされません。



ジ・エンド

久しぶりにブロックくずしをやってみたのですが、やはりなかなか難しいものです。BALLが6個あるのに1面消すのはけっこう大変で、特にブロックがあと1つなのに、BALLが次々となくなっていくときなどはムカついてきます。

それに、反射器とのタイミングがうまくあつてしまうとBALLがブロックのところまで行ってくれず、反射器とピンポンをしている感じになります。ブロックの残り方とボールの位置と反射器の動きがちょうどいいタイミングになると、BALLが同じところを無限に行ったりきたりしているということも起こり得ます。

起きてしまったらどうしようもないので **BRAKE** を押してください。あまりにもしばしば起こる方は(それは消し方が悪いからだと言えそれまでですが……)、8522番地の反射器のスピードを変更してみてください。

このゲームは最初はバグの山で、1回RUNさせるたびにモニタが破壊されてしまい、何をやっても動かなくなり、その都度テープからシステム・プログラムを読み込まなくてはいけませんでした。この点はクリーン・コンピュータというのは不便なものです。ROMのコンピュータならリセットをかけるだけで済むというのに……、やはりクリーン・コンピュータではディスクが必需品みたいですね(早く欲しいなー!!)。

アセンブラがないとやはりかなり大変です。かといって自分でアセンブラを作ろうという気もありませんし、早く誰か発表してくれないかと心待ちにしています。

このゲームで筆者のハイスコアはやっとこさ3,450点でした(あと1個で2面が消えたのに)。

リスト1

0000 DBE8	V-RAMON: IN	A, (0E8H)
0002 CBFF	SET	7A
0004 D3E8	OUT	(0E8H), A
0006 3E02	LD	A, 2
0008 D3F4	OUT	(0F4H), A
000A C9	RET	

リスト2

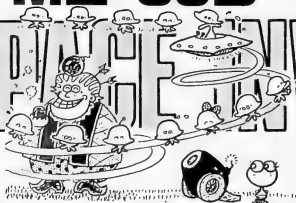
0000 DBE8	KEYIN: IN	A, (0E8H)
0002 E6F0	AND	0F0H
0004 B1	OR	C
0005 D7E8	OUT	(0E8H), A
0007 DBEA	IN	A, (0EAH)
0009 C9	RET	

リスト3 BLOCKくずしメモリ・ダンブ

8000 05 01 10 27 27 35 80 01 1490	8200 40 80 00 00 00 00 00 02 1C2	8570 99 81 00 00 00 00 00 00 3A 154	8828 18 80 76 01 07 00 00 00
8008 08 03 10 27 35 80 01 1490	8208 40 02 01 00 00 00 00 00 00 107	8578 80 00 FE 01 00 00 00 00 00 12D	8830 CE 00 00 EC 85 C3 80 00 163
8010 00 35 00 00 00 00 00 00 00 107	8208 00 01 02 04 00 00 00 00 00 107	8580 0E 12 CD 00 00 00 00 00 00 14E	8838 CB 00 00 CA CD EC 85 C3 80 163
8018 80 70 C6 30 12 13 7E 30 16A	8208 40 80 00 00 00 00 00 00 02 1C2	8588 1C FF 77 70 7D FE 18 CB 179	8840 87 28 7E FE 21 28 70 FE 16E
8020 12 06 05 81 7E FE 30 C0 16A	8208 04 02 01 00 00 00 00 00 00 107	8590 7E FF CA CC 81 E5 CD 14E	8848 7F 28 7E FE 93 CD 26 87 FE 16E
8028 00 00 00 00 00 00 00 00 00 129	8208 00 01 02 04 00 00 00 00 00 107	8598 F6 B1 E1 25 00 00 00 00 00 197	8858 FE 55 20 06 2B CD F8 17E
8030 00 00 00 00 00 00 00 00 00 129	8208 00 01 02 04 00 00 00 00 00 107	85A0 20 00 7E FE FF CB 00 C0 1FA	8860 18 CB 28 7E FE F8 28 CB 17E
8038 70 ED 42 38 05 C7 3E 1AA	8208 00 01 02 04 00 00 00 00 00 107	85B0 81 C7 FA B1 7E FE FF 20 156	8868 FE 55 28 CB 26 87 FE 16E
8040 00 00 00 00 00 00 00 00 00 188	8208 00 01 02 04 00 00 00 00 00 107	85B8 F5 28 B6 B1 7E FE FF 20 156	8870 35 00 CD C5 00 00 00 00 16A
8048 00 18 74 24 80 3E FE 1E 136	8208 00 01 02 04 00 00 00 00 00 107	85C0 00 00 00 00 00 00 00 00 156	8878 96 C5 E4 87 00 00 00 00 00 16A
8058 00 00 00 00 00 00 00 00 00 12C	8208 00 01 02 04 00 00 00 00 00 107	85C8 00 00 00 00 00 00 00 00 156	8880 00 CD CB 00 56 2B 37 00 163
8060 05 24 82 80 54 50 19 19 16A	8208 00 01 02 04 00 00 00 00 00 107	85D0 00 00 21 F1 E8 36 98 0E 01E	8888 7E 03 30 47 2B 1E 28 7E 16E
8068 00 67 11 81 00 19 22 135	8208 00 01 02 04 00 00 00 00 00 107	85D8 1C CD C5 85 00 00 18 ED 18D	8898 FE FF CA CC 81 E5 CD 14E
8070 00 00 00 00 00 00 00 00 00 188	8208 00 01 02 04 00 00 00 00 00 107	85E0 00 00 00 00 00 00 00 00 156	8900 34 02 C3 E4 87 00 CD 00 163
8078 00 00 CD 5F 80 88 08 9B 164	8208 00 01 02 04 00 00 00 00 00 107	85E8 00 00 00 00 00 00 00 00 156	8908 9E CD FD 05 36 E7 84 30 163
8080 18 FB 72 75 00 00 00 00 00 1FA	8208 00 01 02 04 00 00 00 00 00 107	85F0 00 00 00 00 00 00 00 00 156	8910 32 E7 84 CA 30 32 4F 82 16E
8088 00 00 05 67 00 00 00 21 180	8208 00 01 02 04 00 00 00 00 00 107	85F8 00 00 00 00 00 00 00 00 156	8918 C9 00 00 00 00 00 00 00 163
8090 00 00 00 00 00 00 00 00 00 188	8208 00 01 02 04 00 00 00 00 00 107	8600 00 00 00 00 00 00 00 00 156	8920 00 C3 86 00 7E 02 3C 16E
8098 03 20 01 AF 80 80 80 80 80 1C0	8208 00 01 02 04 00 00 00 00 00 107	8608 00 00 00 00 00 00 00 00 156	8928 C3 E5 86 00 00 00 00 00 163
8100 00 00 00 00 00 00 00 00 00 100	8208 00 01 02 04 00 00 00 00 00 107	8610 00 00 00 00 00 00 00 00 156	8930 5A 2B 07 00 CD 00 9A 16E
8108 06 14 77 23 10 FC 9 06 18F	8208 00 01 02 04 00 00 00 00 00 107	8618 C9 00 00 00 00 00 00 00 156	8938 C3 86 00 7E 03 3C 16E
8110 00 00 00 00 00 00 00 00 00 100	8208 00 01 02 04 00 00 00 00 00 107	8620 20 46 27 2D 23 4E 52 1C5	8940 15 2B 0A 23 7E FE 91 28 16E
8118 00 00 00 00 00 00 00 00 00 100	8208 00 01 02 04 00 00 00 00 00 107	8628 20 46 27 2D 23 4E 52 1C5	8948 04 FE 20 0A 00 00 00 00 100
8120 1A 77 23 10 FC 9 06 18F	8208 00 01 02 04 00 00 00 00 00 107	8630 20 46 27 2D 23 4E 52 1C5	8950 84 CD EC 85 C3 80 00 163
8128 00 00 00 00 00 00 00 00 00 100	8208 00 01 02 04 00 00 00 00 00 107	8638 20 46 27 2D 23 4E 52 1C5	8958 FF 28 AA FE 70 28 8A 28 16E
8130 1A 77 23 10 FC 9 06 18F	8208 00 01 02 04 00 00 00 00 00 107	8640 20 46 27 2D 23 4E 52 1C5	8968 00 7E 28 7E FE 0F 28 80 16E
8138 19 CD 41 81 19 CD 35 81 16A	8208 00 01 02 04 00 00 00 00 00 107	8648 20 46 27 2D 23 4E 52 1C5	8978 7F 20 0E 28 CD 21 89 16E
8140 19 CD 41 81 19 CD 35 81 16A	8208 00 01 02 04 00 00 00 00 00 107	8650 20 46 27 2D 23 4E 52 1C5	8980 00 00 00 00 00 00 00 00 163
8148 19 CD 41 81 19 CD 35 81 16A	8208 00 01 02 04 00 00 00 00 00 107	8658 20 46 27 2D 23 4E 52 1C5	8988 00 00 00 00 00 00 00 00 163
8150 00 00 00 00 00 00 00 00 00 100	8208 00 01 02 04 00 00 00 00 00 107	8660 20 46 27 2D 23 4E 52 1C5	8990 00 00 00 00 00 00 00 00 163
8158 0A 36 AA 25 36 55 23 10 CB	8208 00 01 02 04 00 00 00 00 00 107	8668 20 46 27 2D 23 4E 52 1C5	8998 00 00 00 00 00 00 00 00 163
8160 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8670 20 46 27 2D 23 4E 52 1C5	9000 00 00 00 00 00 00 00 00 163
8168 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8678 20 46 27 2D 23 4E 52 1C5	9008 00 00 00 00 00 00 00 00 163
8170 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8680 20 46 27 2D 23 4E 52 1C5	9010 00 00 00 00 00 00 00 00 163
8178 00 99 81 21 1A 64 CD 99 170	8208 00 01 02 04 00 00 00 00 00 107	8688 20 46 27 2D 23 4E 52 1C5	9018 00 00 00 00 00 00 00 00 163
8180 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8690 20 46 27 2D 23 4E 52 1C5	9020 00 00 00 00 00 00 00 00 163
8188 01 21 CE ED 22 49 81 3E 1E7	8208 00 01 02 04 00 00 00 00 00 107	8698 20 46 27 2D 23 4E 52 1C5	9028 00 00 00 00 00 00 00 00 163
8190 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8700 20 46 27 2D 23 4E 52 1C5	9030 00 00 00 00 00 00 00 00 163
8198 01 11 9F 8C 86 80 91 13C	8208 00 01 02 04 00 00 00 00 00 107	8708 20 46 27 2D 23 4E 52 1C5	9038 00 00 00 00 00 00 00 00 163
8200 00 00 00 00 00 00 00 00 00 163	8208 00 01 02 04 00 00 00 00 00 107	8710 20 46 27 2D 23 4E 52 1C5	9040 00 00 00 00 00 00 00 00 163
8208 47 82 21 0E E5 CD 86 80 1E2	8208 00 01 02 04 00 00 00 00 00 107	8718 20 46 27 2D 23 4E 52 1C5	9048 00 00 00 00 00 00 00 00 163
8210 21 52 E6 05 CD 86 80 1E2	8208 00 01 02 04 00 00 00 00 00 107	8720 20 46 27 2D 23 4E 52 1C5	9050 00 00 00 00 00 00 00 00 163
8218 20 CD 86 80 21 16 E5 CD 10F	8208 00 01 02 04 00 00 00 00 00 107	8728 20 46 27 2D 23 4E 52 1C5	9058 00 00 00 00 00 00 00 00 163
8220 47 82 21 0E E5 CD 86 80 1E2	8208 00 01 02 04 00 00 00 00 00 107	8730 20 46 27 2D 23 4E 52 1C5	9060 00 00 00 00 00 00 00 00 163
8228 21 00 8E 00 CD 47 82 21 0A 14A	8208 00 01 02 04 00 00 00 00 00 107	8738 20 46 27 2D 23 4E 52 1C5	9068 00 00 00 00 00 00 00 00 163
8230 EB CD 47 82 21 0A 14A	8208 00 01 02 04 00 00 00 00 00 107	8740 20 46 27 2D 23 4E 52 1C5	9070 00 00 00 00 00 00 00 00 163
8238 06 06 08 05 CD D1 80 E1 16D	8208 00 01 02 04 00 00 00 00 00 107	8748 20 46 27 2D 23 4E 52 1C5	9078 00 00 00 00 00 00 00 00 163
8240 52 82 00 CD 47 82 21 0A 14A	8208 00 01 02 04 00 00 00 00 00 107	8750 20 46 27 2D 23 4E 52 1C5	9080 00 00 00 00 00 00 00 00 163
8248 00 40 82 21 0A 14A	8208 00 01 02 04 00 00 00 00 00 107	8758 20 46 27 2D 23 4E 52 1C5	9088 00 00 00 00 00 00 00 00 163
8250 88 80 05 11 2B 00 19 D1 130	8208 00 01 02 04 00 00 00 00 00 107	8760 20 46 27 2D 23 4E 52 1C5	9090 00 00 00 00 00 00 00 00 163
8258 00 00 00 00 00 00 00 00 00 109	8208 00 01 02 04 00 00 00 00 00 107	8768 20 46 27 2D 23 4E 52 1C5	9098 00 00 00 00 00 00 00 00 163
8260 00 00 00 00 00 00 00 00 00 109	8208 00 01 02 04 00 00 00 00 00 107	8770 20 46 27 2D 23 4E 52 1C5	9100 00 00 00 00 00 00 00 00 163
8268 00 02 04 44 80 00 00 20 10F	8208 00 01 02 04 00 00 00 00 00 107	8778 20 46 27 2D 23 4E 52 1C5	9108 00 00 00 00 00 00 00 00 163
8270 48 11 03 00 00 00 00 01 01 5E	8208 00 01 02 04 00 00 00 00 00 107	8780 20 46 27 2D 23 4E 52 1C5	9110 00 00 00 00 00 00 00 00 163
8278 12 22 42 42 01 00 00 10 1E9	8208 00 01 02 04 00 00 00 00 00 107	8788 20 46 27 2D 23 4E 52 1C5	9118 00 00 00 00 00 00 00 00 163
8280 00 00 80 40 20 10 8E 1D4	8208 00 01 02 04 00 00 00 00 00 107	8790 20 46 27 2D 23 4E 52 1C5	9120 00 00 00 00 00 00 00 00 163
8288 02 00 00 00 00 00 00 00 00 107	8208 00 01 02 04 00 00 00 00 00 107	8798 20 46 27 2D 23 4E 52 1C5	9128 00 00 00 00 00 00 00 00 163
8290 02 01 00 00 00 00 00 10 2A2	8208 00 01 02 04 00 00 00 00 00 107	8800 20 46 27 2D 23 4E 52 1C5	9130 00 00 00 00 00 00 00 00 163
8298 20 10 08 07 00 00 00 10 7F	8208 00 01 02 04 00 00 00 00 00 107	8808 20 46 27 2D 23 4E 52 1C5	9138 00 00 00 00 00 00 00 00 163
8300 00 23 10 08 07 00 00 10 7F	8208 00 01 02 04 00 00 00 00 00 107	8810 20 46 27 2D 23 4E 52 1C5	9140 00 00 00 00 00 00 00 00 163
8308 20 20 12 02 00 00 00 10 7E	8208 00 01 02 04 00 00 00 00 00 107	8818 20 46 27 2D 23 4E 52 1C5	9148 00 00 00 00 00 00 00 00 163
8320 00 00 00 80 40 20 10 2A0	8208 00 01 02 04 00 00 00 00 00 107	8820 20 46 27 2D 23 4E 52 1C5	9150 00 00 00 00 00 00 00 00 163

MZ-80B

SPACE INVADERS



スペース・インベーダー

■FALCON・S

(注) このゲームには、グラフィックV-RAM1が必要で、

あのすさまじいばかりのインベーダー狂騒曲が、一段落してもう何年かたちますが、ゲーム・センターの片隅にまだ残っている所もあります。マイコン界においてもインベーダーはまだ人気があり、ほとんどすべての機種用のものがあるようですが、まだMZ-80B用がありません。

そこでドット・グラフィックを使って、できるだけ本物に近いインベーダーを作ってみました。

1 遊び方

ビーム砲はテン・キーの **[4]** と **[6]** で左右に動き、指を離せばそこに止まっています。ビームの発射は **[スペース]** キーで連続してビームが出てしまわないようにしてあるので、次のビームは一度キーを離してからまた押すようにしてください。

プログラムのスタートは9000番地からです。スタートするとまず説明文が表示され、**[CR]** キーを押すとゲームが始まります。ゲームの内容はあまりにも有名ですから説明の必要はないと思いますが、ようするにミサイルを避けるながらインベーダーを全部やっつけなければならないのです。高得点のコツは後で述べます。

一面消し終わると次の面は前より一段下がった所から開始されます。第11面ではインベーダーは壁と接して出現するので恐怖です。

インベーダーの得点は、1行(一番上)が30点、2、3行が20点、4、5行が10点です。UFOの得点は50、100、150、200、250、300点のどれかで、画面上に表示されます。UFOは一定時間ごとに出現しますが、インベーダーが7匹以下になるともう出てきません。

ビーム砲の数は4台です(本当は3台ですが、1台おまけ)、1,500点を越えると1台追加されます。ビーム砲がすべてやられるとゲーム・オーバーです。またインベーダーが一番下の行まで来ると侵略されたことになり、ビーム砲が残っていてもゲーム・オーバーになってしまいます。

ハイスコアを0にしたいときは、A 6 F E 番地と A 6 F F 番地の値を00にしてください。

2 プログラムの入力

3 チェック・サムのプログラム

今回マシン語で8バイトごとのチェック・サムを見るプログラムを作りましたので、そのアセンブル・リストをリスト1に載せておきます。8100番地からスタートさせると、S-ADR. \$と出ますからスタート・アドレス(9000)を入れ、次にE-ADR. \$にエンドアドレス(A700)を入れると、ダンプリストとともにその右端にチェック・サムの値が、表示されます。**[スペース]** キーを押すと、その間、画面は静止します。**[BREAK]** キーで中断されます。このプログラムはリロケータブルですから、どのアドレスから打ち込んでもかまいません。

4 プログラムについて

55匹のインベーダーを周囲の条件を調べながら動かすというのはマシン語でもかなり重荷で、特にドット・グラフィックでインベーダーを描くのに時間をとられるため、インベーダーの移動時にはビーム砲の動きがつかづよう感じられるかもしれません。インベーダーの数が減ってくると、だんだんスムーズに動くようになります。

実際の判定はワーク・エリア内に40×25の仮想画面を作り、その中でやっています。この内容をドット・グラフィック画面に投影しているという感じになります。

インベーダーの行進音をつけたため、ますます動きがギコチンになってきている感じもありますが、この行進音があるかないかでゲームに大きな差が出ます。インベーダーが、少なくなってくると音は連続して出るため、この音の出る時間がインベーダーの動きを規定してしまうことに

[illegible]

図1 ジェネラル・フローチャート



図4 ビーム発射③

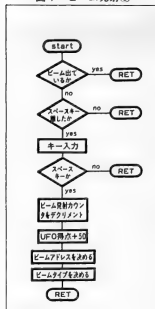


図5 ビーム上昇④

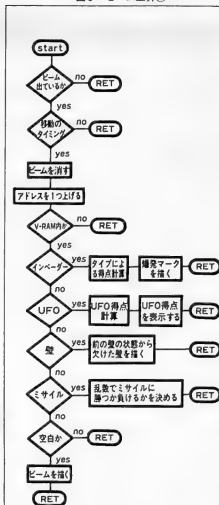


図6 ミサイル降下⑤

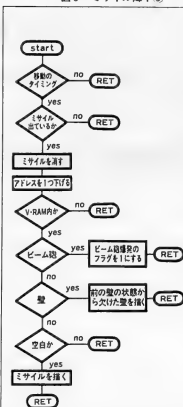


表1 インバーダー数と速度の関係

インバーダーの数	速度
55～32	64
31～16	32
15～8	16
7～5	8
4～3	4
2～1	2

ルーチン①～⑤のフローチャートを図2～図6に示します。

インバーダーは数が少なくなるにつれて速くなり、その関係を表1に示します。UFOの内容は実際にプログラム中でタイミングを取るときに使われる値で、小さいほど速くなります。インバーダーは縦の列を単位として管理していて、11列分のワーク・エリアを持ち、その列すべてのインバーダー（5匹）が消えるとワーク・エリアに対応するデータが、0になるようにしています。よって、インバーダーの数が減ってくると、インバーダーのいない列はサーチしなくてもいいので時間の節約になります。

ビームは一発だけで画面に出ているのが、消えるまで次のビームは出ません。UFOの内容はビーム発射ごとに50点ずつ増加し、250点の次は50点というように回転しています。本当はもっと複雑な周期にしたかったのですが…、またビーム発射ごとにカウンタが減っていき、これが0のときにUFOに当たると300点が得られるようにしてあります。

ミサイルは画面上、最高3発で、インバーダーの下にビーム砲があるときは、一発は狙って落ちてきます。他の2発はランダムな場所から発生します。ミサイルはインバーダーの行から1行あけて発生します。これにより名古屋警が可能となるわけです。

壁はミサイルやビームが当たるとだんだんにくずれていくので、欠けた形を30種類作り、そのときの条件によって形を選ぶようにしています。



細かいマシン語ルーチンの内容を表2に、ワーク・エリアの内容を表3に、データ・エリアの内容を表4に示しますので参考にしてください。

5 高得点のコツ

インバーダー・ゲームの高得点の方法はいろいろと有名な方法があるようですが、私の知っている程度のことを書きます。何しろ私は本物のインバーダーでは、せいぜい5,000点くらいしかいったことがありませんので、ここに書いたことも信用しない方がいいかもしれません。

①インバーダーは端へ来ると一段下がるので、少しでも降りてくる時間を遅らせるよう、最初のうちは縦の列を単位

に消していきます。

② UFOの得点はスタートしてから23回目のビームで、UFOを撃ち落とすと300点が得られます。その次からは15回目ごとに300点になるので、撃った数をよく覚えておいて300点UFOをものにしていきます。

③ UFOはどちらから出てくるか乱数で決まりますから、どこから来てもいいようにビーム砲の位置を動かしておきます。もっともこのUFOは速いのでは追いつけず追いつきませんが、こういうときはUFOばかり見ているので、ミサイルにやられやすくなります。

④ インベーターは数が減ると速くなっていくので、最初のうちはあまり数を減らさないようにします。ビーム発射数を合わせるのには、空撃ちや壁をくすしたりします。とくに最後の速くなったときにはかえって壁はじゃまになることも多いので、消してしましましょう。

⑤ UFOはインベーターが7匹以下になると出てきませんから、インベーターが8匹になったら危険と思われる所に降りてくるまで放っておき、その間、UFOに専念します。

⑥ 画面が進んでインベーターが、下の方に出現するようになってきたら名古屋撃ちを使いましょう。私はこれがヘタクソなのでこれ以上は申しません。

6 ゲームの変更点

このゲームは本物より少し易しくしてあるため、つまらないという方もいるかもしれませんが、このゲームの変更点をあげておきます。

① UFOスピードは9EEA番地(07)を変えます。

② ミサイルスピードは9C9C番地(07)を変えます。

③ ミサイル1(ビーム砲を狙ってくるもの)発生率は9B85番地(0F)を、ミサイル2(ランダム・ミサイル)発生率は9BF0番地(1F)を変えます。

④ ビーム・スピードは9857番地(01)を変えます。これらの変更する数値は、00, 01, 03, 07, 0F, 1F, 3F……という値にしてください。

7 おわりに

このインベーターは、もうちょっとで目の見えないところでした。たった一箇所のバグに3日間も泣かされて、いいかげんイヤになってボツにしようと思ったとき、ヒョイと見つかりました。これが何と初歩の初歩のミスで、16進数では19の次は1Aになるのにこれを20にしていたためでした。こんなミスをするわけがないと信じているので、他の場所を必死に直していたわけです。

MZはシステムがRAM上なので、バグの所へ来たたびにしシステムまで壊れてしまうのには閉口しました。マシン語のバグ取りのいい方法はないものではないでしょうか。

表2 マシン語ルーチン

アドレス	内 容
9000	メイン・ルーチン
905B	メモリ後半をV-RAMにする。
9069	キー入力サブ
9076	8×8のドットを消すサブ
9089	キャラクタ画面をクリアする。
9095	ワーク・エリア仮想画面に0を入れる。
909C	ドット画面をクリアする。
90A6	ハイ・スコアの判定
90CA	10進出力サブ
9108	UFO得点用10進出力サブ
911A	タイマ
9129	乱数発生サブ
914D	メイン・クロック、3進クロックの転換 【スペース】 キーを離したかの判定
9173	ビーム砲追加の判定
919E	各種プリント・サブ
91DD	アドレスを1行下げる。
91E7	カーソルの点滅と 【CR】 キー入力待ち
9209	キャラクタ画面上のアドレス計算
9216	ドット画面上の "
9223	ワーク・エリア仮想画面上のアドレス計算
9258	HLレジスタの内容をAレジスタに入れる。
9261	キャラクタおよびドット・プリント・サブ
92B5	ビームを描く
92D5	ミサイルを描く
9318	インベーターを描く
9378	ビーム砲を描く
9398	UFOを描く (type 1)
93A8	インベーター爆発マークを描く
93BE	ビーム砲爆発マークを描く
93C9	UFOを描く (type 2)
93E7	壁を描く
940A	欠けた壁を描く
9427	キャラクタおよびドットのクリア
944E	初期画面 (インベーター、壁、文字、ビーム砲を描く)
9526	インベーター移動
968D	インベーター行進音カウンタのインクリメント
96A0	端にきたかの判定
96D0	インベーター数によりインベーターの速度を決める。
96F3	インベーターを1行下げ、侵略かの判定
9747	ビーム砲移動
97A6	インベーター行進音を出す。
97F9	ビーム発射
984D	ビーム上昇
9AE9	得点の計算・表示
9B7F	ビーム砲を狙って落ちてくるミサイルの発生
9BE1	ランダム・ミサイル発生
9C96	ミサイル降下のメイン
9CBE	ミサイル降下のサブ
9E93	UFOの発生
9EDF	UFOの移動
9F4C	UFO得点の表示とクリア
9F7C	ビーム砲爆発とスタート位置への表示
9FCD	ゲーム・オーバー処理
A031	面終了時の処理
A055	説明文の表示
A0F3	第1初期設定
A116	第2初期設定

本物のインベーターには、かなり大金を使ってしまい、根みを持っているので、このゲームは全体的に少し易しくなっています。それでも、12,000点くらいはいきませんでした。

【参考文献】

- 1) 近藤洋一・近藤康司: "SPACE INVADERS", I/O, '79年7月号
- 2) Mr. MX: "スペースインベーターを解制するが", I/O, '79年8月号
- 3) "インベーターマシン大研究", マイコン, '79年8月号

表3 ワーク・エリア

アドレス	内 容	アドレス	内 容
A6FE, A6FF	ハイ・スコア	A737, A738	ミサイル1のアドレス
A702, A703	乱数値	A739	ミサイルの種類
A704	アドレス計算時のスタック	A73A	ミサイルのタイプ
A706	メイン・クロック値	A73B~A744	ミサイル2, ミサイル3で上と同様
A708	3進クロック値 (ビーム砲用)	A747~A751	インベダーの縦の単位位の存在しているかのフラグ
A709, A70A	スコア	A756	インベダー侵略のフラグ
A70C	ビーム砲を追加したかのフラグ	A757~ACCF	仮想TV画面 (40×35)
A70D	ビーム砲の数		
A70F	スペースキーを離したかのフラグ		
A710, A711	ビーム砲アドレス		
A713	インベダー・スピード		
A714	インベダー移動方向		
A715	インベダーのタイプ		
A716	インベダーの総数		
A717	インベダー行進音のカウンタ値		
A719, A71A	インベダー集団左上のアドレス		
A71C, A71D	インベダー集団左上のアドレスの初期値		
A71F	ビーム砲発射のフラグ		
A720	インベダー行進音を出すかのフラグ		
A721	ビーム砲のタイプ		
A722	ビーム出現しているかのフラグ		
A723, A724	ビームのアドレス		
A725	ビームのタイプ		
A726	ビーム発射数のカウンタ値		
A727, A728	UFOの得点		
A729	UFOが出現しているかのフラグ		
A72A	UFO移動方向		
A72B, A72C	UFOのアドレス		
A72E, A72F	UFO出現までのカウンタ値		
A730	UFOのタイプ		
A731, A732	UFO得点表示アドレス		
A734	UFO得点を表示しているかのフラグ		
A735	UFO得点表示期間のカウンタ値		
A736	ミサイル1が出現しているかのフラグ		

表4 データ・エリア

アドレス	内 容
A184~A19C	ビーム砲追加音
A19E~A1B9	ビーム砲出現音
A1BA~A1C5	インベダー爆発音
A1C6~A20C	UFO消滅音
A20D~A24C	ビーム砲爆発音
A24D~A261	ゲーム・オーバー時の音
A262~A289	一面終了時の音
A28A~A2AD	4, 5行目のインベダー
A2AE~A2D0	2, 3行目のインベダー
A2D1~A2F4	1行目のインベダー
A2F5~A318	インベダー爆発マーク
A319~A33C	ビーム砲
A33D~A35F	ビーム砲爆発マーク
A360~A372	ビーム
A373~A396	ミサイル
A397~A3C2	UFO
A3C3~A3DC	壁の上の行
A3DD~A3F6	壁の下の行
A3F7~A4EA	欠けた壁 (30種類)
A4EB~A56B	画面右側の枠と文字
A56C~A5EB	ゲーム・オーバー時の枠と文字
A5EC~A6C5	説明文

ダンブ・リスト

9000 CD 5B 90 3E 07 CD 50 0E 12B	9170 00 00 00 2A 09 A7 3A 0C 120	9260 8B 92 D1 C1 C9 C5 D5 3E 150
900B CD F3 A0 CD 16 A1 CD 9C 14D	917B A7 A7 CB 7D FE 9A 0B 3A 139	926B 80 CD 69 92 11 7C A3 CD 152
9010 90 CD 89 90 CD 80 9A CD 154	9180 0D A7 3C 32 0D A7 C6 3A 1CC	9270 8B 92 D1 C1 C9 C5 D5 3E 150
901B 73 A0 CD 4E 9A CD 4D 91 16D	918B 32 F4 D2 AF 32 0C A7 3E 1CA	927B 8E CD 69 92 11 85 A3 CD 15C
9020 CD 26 95 CD 47 97 CD 9F 1F9	9190 07 CD 50 0E 11 84 A1 CD 135	9300 8B 92 D1 C1 C9 C5 D5 3E 150
902B 97 CD 4D 98 CD 7F 9B CD 1FD	919B 3F 0F C9 00 00 21 00 13B	930B 8F CD 69 92 11 8E A3 CD 166
9030 E1 9B CD 6E 9C CD 93 9E 179	91A0 D0 1A A7 CB 77 13 23 18 11E	9310 8B 92 D1 C1 C9 00 00 00 17B
903B CD DF 9E CD 4C 9F 3A 1F 15B	91AB FB 06 0A 1A 77 23 13 10 1DF	931B C5 D5 3E 65 CD 71 92 11 1E
9040 A7 FE 01 CA 7C 9F 3A 16 1DB	91B0 FA C9 0A 28 AF 77 23 10 14A	9320 8A 82 CD 93 92 D1 C1 C9 179
904B A7 FE 00 DA 31 A0 CD 42 16F	91B8 FC C9 21 00 F3 1A A7 CB 162	932B C5 D5 3E 67 CD 71 92 11 120
9050 05 CA 00 00 1B C7 00 00 1AE	91C0 77 13 09 1B FB 00 00 1A3	9330 9C 82 CD 93 92 D1 C1 C9 18B
905B 00 00 00 DB EB CB FF D3 160	91CB 06 C5 77 19 10 FC C9 0A 13B	933B C5 D5 3E 69 CD 71 92 11 122
906B E8 3E 02 D3 F4 C9 00 00 1B8	91D0 0B 1A 77 CD 90 D1 13 10 1F7	9340 AE 82 CD 93 92 D1 C1 C9 19D
906B 00 DB EB 6E F0 B1 D3 EB 105	91DB FB C9 00 00 00 D5 11 2B 1CF	934B C5 D5 3E 6B CD 71 92 11 124
9070 DB EA C9 00 00 00 D5 14B	91E0 00 19 D1 C9 00 00 00 21 14A	9350 0A 82 CD 93 92 D1 C1 C9 16F
907B C5 06 0B AF 11 2B 00 CD 1B8	91EB A0 D3 3A 1E 1A 1E CD 1EC	935B C5 D5 3E 6D CD 71 92 11 126
9080 CA 91 C1 E1 D1 C9 00 00 197	91F0 91 CD 32 0B FE 0D CB CD 13B	9360 D1 A2 CD 93 92 D1 C1 C9 1C0
908B 00 21 00 D0 0E 19 CD 82 197	91FB 62 05 CA 00 3A 20 1A 19D	936B C5 D5 3E 6F CD 71 92 11 12B
9090 91 0D 20 FA C9 21 57 A7 1A0	9200 1E CD 1C 91 1B 0A 00 00 194	9370 E3 A2 CD 93 92 D1 C1 C9 1D2
909B 0E 23 1B F2 21 00 00 14A	920B 00 E5 21 00 D0 22 04 A7 1A3	937B C5 D5 3E 7B CD 71 92 11 131
90A0 CB 1B EB 00 00 3A 0A 10F	9210 E1 11 2B 00 1B 29 E5 21 161	9380 19 A3 CD 93 92 D1 C1 C9 109
90AB A7 47 3A FF A6 8B 2B 03 1B0	921B 00 E0 22 04 A7 E1 11 40 1D2	938B C5 D5 3E 79 CD 71 92 11 132
90B0 D0 1B 09 3A 09 A7 47 3A 15C	9220 01 18 1C E5 21 57 A7 22 15B	9390 2B A3 CD 93 92 D1 C1 C9 11B
90BB FE A6 8B 2A 19 A7 22 12B	9230 CB 38 05 DA DB A7 1B 07 139	939B C5 D5 3E 7B CD 71 92 11 14F
90C0 FE A6 11 50 D2 1B 03 00 1F2	923B FE 2B 38 03 06 2B 67 15B	93A0 D1 A3 CD 93 92 D1 C1 C9 1A1
90CB 00 00 D5 01 10 27 CD FB 1D2	9240 E5 05 45 2A 04 A7 FE 01 17B	93AB C5 D5 11 F5 A2 CD 93 92 134
90D0 90 01 EB 03 CD FB 90 01 1D2	924B 2B 04 7D 19 10 FD C1 5B 170	93B0 D1 C1 C9 C5 D5 11 07 A3 1B0
90DB 64 00 CD FB 90 01 0A 00 1C4	9250 16 00 19 C1 C9 00 00 189	93BB CD 93 92 D1 C1 C9 C5 D5 1E7
90E0 CD FB 90 01 0A 00 13 13 1ED	925B E5 D5 CD 23 92 7E D1 E1 16C	93CB 11 3B A3 CD 93 92 D1 C1 17B
90EB 3E 30 12 06 05 E1 7E FE 15B	926B C9 E5 CD 23 92 3A 00 E1 147	93CB C9 C5 D5 11 4F A3 CD 93 16A
90F0 30 CD 3A 00 23 10 F7 C9 119	926B C9 E5 CD 23 92 3A 00 E1 147	93DB 92 D1 C1 C9 00 00 C5 182
90FB C5 A7 3E 30 ED 42 3B 03 144	9270 09 E5 F5 CD 23 92 3A 00 E1 147	93DB C5 3E 9B CD 7D 92 11 97 12F
9100 3C 1B F9 C1 09 12 13 C9 105	927B D1 E5 CD 23 92 3A 00 E1 147	93EB A3 CD A4 92 D1 C1 C9 C5 1C6
910B 01 0A 00 00 00 00 00 12C	928B 23 3C 77 E1 C9 E5 F5 CD 127	93EB D5 3E 1F CD 7D 92 11 C3 122
9110 30 12 13 3E 30 12 09 00 19E	928B 23 3C 77 E1 C9 E5 F5 CD 127	93FB A3 CD A4 92 D1 C1 C9 C5 1C6
911B 00 00 16 05 1E 00 1B 7A 1CE	928B 00 00 00 E5 D5 CD 16 92 12F	93FB D5 3E 22 CD 7D 92 11 D0 1FF
9120 83 20 FB C9 00 00 00 00 197	929B 01 18 0C E5 D5 CD 16 92 12A	9400 A3 CD A4 92 D1 C1 C9 00 101
912B 00 E5 D5 2A 02 A7 3A 3D 13E	929B D1 E5 CD 23 92 3A 00 E1 147	940B 00 00 C5 D5 CD 69 92 E5 147
9130 19 19 7B 64 67 11 81 00 12C	92A0 CF 91 E1 C9 E5 D5 1A 1A7	9410 A7 21 F7 A3 11 0B 00 10 12B
913B 19 22 02 A7 7C D1 E1 C9 1D0	92AB 92 D1 E5 CD 23 92 3A 00 E1 147	941B 0B EB E1 CD 8B 92 D1 C1 151
9140 CD 29 91 8B 0B 9B 1B FB 1C2	92AB 1B 07 00 00 C5 D5 3E 1D7	9420 C9 19 1B F3 00 00 C5 1B2
914B 00 00 00 00 00 21 0A 07 1E	92BB 82 CD 69 92 11 A1 A3 CD 12C	942B D5 00 A1 61 E5 CD 16 92 1EF
9150 3A 3A 0B 47 3C FE CD 20 17A	92CB 8B 92 D1 C1 C9 C5 D5 3E 150	9430 CD 76 90 D1 C1 C9 E5 1F4
915B 01 AF 32 0B A7 3A 0F 1B1	92CB 83 CD 69 92 11 A4 A3 CD 135	943B CD 27 8A 24 CD 27 9A E1 115
9160 A7 CB 0E 13 CD 69 90 CB 121	92DB 8C 92 D1 C1 C9 C5 D5 3E 150	9440 C9 E5 CD 27 9A 24 CD 27 9A 151
916B 4F CB AF 32 0F A7 C9 00 177	92DB 8C CD 69 92 11 73 A3 CD 14B	944B 94 E1 C9 00 00 00 CD 8B 194

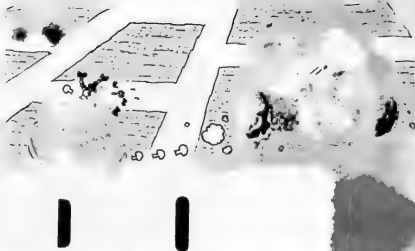
9450 90 CD 9C 90 CD 95 90 2A 1A5
9458 1C A7 22 19 A7 E5 06 0B 198
9460 CD 58 93 24 24 10 F9 E1 1EA
9468 2C 0C 02 06 08 E5 CD 128
9470 28 93 24 24 10 F9 E1 1EA
9478 2C 00 20 F0 0E 02 06 0B 1A6
9480 E5 CD 18 93 24 24 10 F9 1EA
9482 E5 2C 2C 00 20 F0 21 16 18D
9490 05 06 04 CD E7 93 7C C6 198
9498 06 67 10 F7 21 17 05 0E 1EA
94A0 04 CD F7 93 7C C6 06 10A
94B0 10 F7 CD B0 94 CD FD 94 16C
94B2 21 1F 00 11 EB 4A CD 15 192
94B8 95 21 E7 00 16 85 CD 106
94C0 15 95 21 D7 D1 11 14 13F
94C8 CD 15 95 21 01 D2 11 65 E1
94D0 AS CD 25 92 97 D2 11 16F
94D8 AS CD 11 91 21 C7 09 A7 13B
94E0 11 60 D1 CD CA 90 2A FE 191
94EB 86 11 50 D2 CD CA 90 3A 13A
94F0 0D A7 C6 30 32 F4 D2 21 1C3
94F8 13 21 C3 8B 93 21 19 05 51
9500 22 10 A7 CD 8B 93 3E 25 104
9508 32 21 A7 11 9E A1 CD 3F 156
9510 CF 09 00 00 00 06 05 E5 1CB
9518 CD A1 91 13 E1 CD 00 91 12E
9520 10 F5 CD 00 00 00 3A 06 10E
9528 A7 47 3A 13 A7 CD 06 1EA
9530 97 CD 80 96 3E 01 32 20 11B
9538 A7 3A 14 A7 2B CD 0C 14A
9540 A4 96 30 10 AF 32 CD 14 116
9548 C3 9A CD 00 00 00 16 1B3
9550 9E 01 18 F1 3A 15 97 FE 13C
9558 01 28 04 3E 01 18 02 3E 1C4
9560 02 32 15 A7 2A 19 A7 FD 1D
9568 21 17 07 06 08 3A 14 17 115
9570 A7 20 2C 3E 15 A7 01 1EA
9578 28 32 E5 24 22 19 A7 E1 126
9580 FD 21 51 A7 7C C6 14 A7 103
9588 FD 0E 00 28 0A E5 16 14F
9590 00 CD 57 94 FD 72 00 E1 10A
9598 25 25 FD 28 10 EA C9 3A 16F
95A0 15 A7 FE 01 20 06 E5 25 1EB
95B8 22 19 A7 E1 FD 7E 00 A7 1E5
95B0 28 0A E5 13 A7 CD 06 155
95B8 FD 72 00 E1 24 24 F7 23 18B
95C0 10 EA C9 00 00 00 0E 05 1D6
95C8 3A 14 A7 2B 2E 3A 15 141
95D0 28 09 A7 E1 28 AE CD 58 92 1D3
95D8 FE 65 20 05 CD 48 93 14A
95E0 10 FE 69 20 05 CD 48 93 14A
95E8 18 07 FE 60 20 05 CD 48 93 14A
95F0 93 16 01 2C 2C 00 20 10C
95F8 09 00 00 CD 58 92 1D3
9600 67 20 05 CD 18 93 10 12C
9608 FE 68 20 05 CD 38 93 18 13E
9610 07 FE 6F 20 05 CD 58 93 151
9618 16 01 2E 18 05 20 20 C9 142
9620 00 00 CD 58 92 1D3
9628 20 09 CD 37 94 25 18 1CB
9630 93 18 18 FE 68 20 09 CD 122
9638 37 94 25 CD 38 93 18 08 18B
9640 FE 6F 20 0A CD 37 94 25 154
9648 CD 58 93 24 16 01 2C 2C 14B
9650 0D 20 00 C9 00 00 0E 10A
9658 05 CD 58 92 FE 65 20 09 148
9660 CD 37 94 24 28 93 18 15C
9668 18 FE 69 20 09 CD 37 94 140
9670 24 CD 48 93 18 0B FE 6D 15A
9678 20 0A CD 37 94 24 CD 68 11B
9680 93 25 16 2C 2C 00 20 154
9688 0D C9 00 00 00 3A 17 23 191
9690 FE 04 20 04 3E 01 18 01 17E
9698 3C 32 17 A7 C9 00 00 1F5
96A0 26 1C 18 02 26 01 3A 19 1D6
96A8 A7 6F 3E 18 02 3E 18 02 1D6
96B0 92 3A 15 A7 FE 01 C8 7E 1CD
96B8 FE 67 28 0F FE 68 28 0B 13B
96C0 FE 6F 28 07 CD 00 91 10 1E7
96C8 EE A7 C9 37 C9 00 00 15E
96D0 3A 16 A7 FE 1F 28 15 14F
96D8 0F 28 11 FE 07 28 0D FE 180
96E0 04 20 04 3E 01 18 05 FE 184
96E8 02 00 3E 01 32 13 A7 C9 18E
96F0 00 00 00 28 19 02 3E 1C1
96F8 22 19 A7 E1 0E 05 E5 1C8
9700 08 CD 58 92 FE 67 20 09 150
9708 CD 37 94 2C 2C 93 18 164
9710 18 FE 68 20 09 CD 37 94 142
9718 2C CD 48 93 18 0B 28 93 15C
9720 20 17 CD 37 94 2C CD 68 150
9728 93 70 FE 19 20 0A 3E 01 101
9730 32 56 A7 32 1F A7 E1 C9 1D1
9738 20 24 10 CA E1 2C 2C 1B2
9740 0D 20 8B C9 00 00 00 1EA
9748 08 A7 FE 01 C0 2A 10 1A7
9750 0E 12 CD 69 90 CD 67 28 140
9758 28 CD 72 00 3A 21 A7 FE 12A
9760 01 28 0B 3E 01 32 3E 16A
9768 C3 78 93 7C FE 1A CB 7E 16B
9770 02 32 21 A7 CD 37 94 2A 18B

9778 22 10 A7 C3 8B 93 00 00 187
9780 00 3A 21 A7 FE 01 2B 0F 138
9788 3E 01 32 21 A7 CD 37 94 1D1
9790 25 22 10 A7 C3 78 93 70 148
9798 FE 02 CB 3E 02 32 21 A7 102
97A0 C3 8B 93 00 00 3A 20 138
97AB FE 01 CA CB 97 3A 16 122
97AB A7 FE 01 CD 16 91 FE 02 177
97B0 02 01 2B 0A 00 3A 16 179
97B8 06 16 02 12 08 16 03 169
97C0 C3 1C 1F AF 32 20 A7 3A 152
97D0 17 A7 FE 01 20 05 21 95 198
97D8 02 18 15 0A 97 93 70 148
97E0 F0 02 18 0C FE 03 20 05 13C
97E8 21 60 03 18 03 21 E5 03 1A7
97F0 01 02 00 C3 22 0F 00 00 1F7
97FB 00 3A 22 A7 A7 CD 3A 0F 1B3
9800 A7 02 0E 13 A7 CD 69 90 1F5
9808 CB AF C0 3E 01 32 0F A7 101
9810 32 22 A7 3A 27 A7 FE 19 11A
9818 28 04 C6 05 18 02 3E 05 154
9820 32 27 3E 02 A7 20 1C6
9828 04 3E 01 18 01 30 32 26 1FE
9830 A7 2A 10 A7 3A 21 A7 FE 188
9838 01 28 07 22 23 A7 3E 02 15E
9840 18 04 24 22 23 A7 32 25 183
9848 97 C9 00 00 00 3A 06 10E
9850 FE 01 C0 3A 06 A7 E6 01 18D
9858 C0 2A 23 A7 CD 58 92 FE 169
9860 82 08 A7 FE 83 20 03 CD 11F
9868 97 94 20 23 19 A7 20 173
9870 20 05 AF 32 22 A7 C9 3A 1D2
9878 25 A7 FE 01 20 51 CD 58 161
9880 92 FE 66 20 07 25 01 01 144
9888 00 C3 FF 9A FE 67 28 95 1EE
9890 02 00 00 FE 68 20 07 25 11E
9898 C3 05 9B FE 6A 20 07 25 117
9900 01 02 00 C3 FF 9A FE 68 1CB
9908 28 05 FE 6C 20 07 25 01 1E4
9910 02 00 C3 05 9B FE 6E 20 127
9918 07 25 01 03 C3 FF 9A 18C
9920 FE 6F 28 05 FE 70 20 3C 14A
9928 25 01 03 C3 05 9B CD 159
9930 58 92 FE 65 28 9B FE 66 189
9938 97 28 07 FE 67 28 FE 67 178
9940 28 14 18 23 FE 9A 28 16 149
9948 97 28 11 FE 99 28 0A 193
9950 FE 9A C2 13 9A 25 CD 02 11E
9958 11 94 C3 4C 9B 00 00 FE 13C
9960 28 7A FE 67 22 28 76 FE 17E
9968 28 72 FE 1F 28 72 FE 173
9970 21 72 FE 1F 28 72 FE 173
9978 01 28 72 9F 18 72 FE 13C
9980 02 28 6E FE 07 28 6A FE 12D
9988 18 28 66 FE 02 28 6A FE 14A
9990 28 62 FE 0E 28 5E FE 122
9998 08 72 FE 0D 28 6E FE 144
99A0 08 56 FE 10 28 56 FE 119
99B0 18 52 FE 1E 28 4E FE 120
99C0 13 28 4E FE 15 28 4A FE 10C
99D0 18 28 4E FE 14 28 52 FE 115
99E0 19 42 FE 1C 28 4E FE 111
99F0 04 28 5A FE 05 28 50 FE 1F9
9998 17 28 4C FE 1A 28 4B C3 1D6
99A0 08 9A 00 00 00 3E 16 18 10
99B0 36 3E 0E 18 32 3E 0E 18 140
99C0 2E 3E 18 2A 3E 07 18 123
99D0 26 3E 06 18 22 3E 09 103
99E0 1E 3E 0E 18 1A 3E 11 18 02
99F0 1E 3E 12 18 13 3E 18 101
99A0 0E 3E 05 18 0E 3E 08 1D3
99B0 0E 3E 15 02 3E 1C 1D 19E
99C0 0A 9A 3F 32 22 A7 C9 1D 1D6
99D0 27 1A F6 3E 1A 18 FE 122
99E0 29 1E 1B 3E 0C E7 1A1
99F0 3C 1E 1B 3E 03 18 DF 180
9A00 3E 1A 0B 3E 18 0D 17 160
9A08 3E 13 0B 3E 1A 18 DF 178
9A10 00 00 00 00 20 28 95 144
9A18 22 0D 01 FE 24 28 CD 130
9A20 23 28 CD 1F 28 CD FE 128
9A28 21 28 CD 1F 28 9A FE 1E5
9A30 01 28 C9 FE 0C 25 1E5
9A38 28 C1 14 28 8D 8D 1E5
9A40 02 28 8D FE 28 89 FE 1C9
9A48 13 28 85 FE 07 28 82 FE 1F0
9A50 0A C1 91 97 FE 15 C1 01 FE
9A58 0E 28 82 FE 09 C4 59 FE 107
9A60 10 28 9D FE 16 C4 99 125
9A68 19 28 98 FE 1C 28 94 1AD
9A70 FE 1E 28 98 FE 1C 28 94 1AD
9A78 FE 1E 28 9D FE 1E 28 94 1AD
9A80 28 97 FE 0A C4 97 99 1D0
9A88 12 C4 E7 99 FE 18 C4 E7 123
9A90 99 FE 1B C4 E7 99 FE 1D 117
9A98 9A E7 99 FE BE 28 99 FE 118

9A00 8C 28 09 FE 00 C0 C3 B5 1F3
9A08 92 00 00 00 CD 99 1E6 1FF
9A10 03 28 06 CD 24 93 B5 135
9A18 92 00 00 CD 29 A1 C2 1E6
9A20 22 47 32 36 A7 C2 93 156
9A28 CD 29 91 E6 03 28 EA CD 14F
9A30 24 9B C3 E5 92 00 00 00 1D9
9A38 98 28 98 28 98 28 98 132
9A40 FE 00 C0 C3 E2 00 00 1D8
9A48 0E E5 2A 09 A7 09 22 09 1F3
9A50 A7 11 60 D1 CD CA 90 CD 1DD
9A58 73 91 E1 C9 00 00 00 E5 133
9A60 98 28 98 28 98 28 98 189
9A68 93 CD 99 9A AF 32 22 A7 18D
9A70 11 BA A1 CD 3F 0F E1 CD 135
9A78 37 94 21 16 A7 35 CD D0 171
9A80 96 00 00 00 06 02 FD 21 18C
9A88 38 47 FD 7E 00 00 28 11 5D
9A90 7D FD BE 01 20 08 7C FD 1DD
9A98 BE 02 20 05 FD 36 00 00 11B
9B00 49 11 05 00 FD 19 10 E2 1E7
9B08 C9 00 00 00 AF 32 22 A7 173
9B10 32 29 A7 3A 26 A7 A7 2D 108
9B18 0E 4B 4B 27 A7 1B 05 01 12B
9B20 1E 00 C3 CD E9 9A CD 09 109
9B28 92 22 31 A7 3E 01 00 1F5
9B30 34 A7 E1 CD 91 C6 1F9
9B38 A1 C3 3F 0F 00 00 00 3A 1E2
9B40 06 A7 C6 02 E6 0F 00 3A 164
9B48 36 A7 A7 00 2A 10 A7 CD 1FC
9B50 9A 06 32 36 A7 C9 0A 10 1D
9B58 97 24 CD B6 9E 00 C8 1AF
9B60 F5 CD 29 91 E6 07 F1 C8 122
9B68 32 3A 21 2C 22 37 A7 18B
9B70 3E 01 32 36 A7 C9 0A 10 1D
9B78 00 00 00 00 00 00 28 18 189
9B80 FE 68 28 14 FE 6F 28 10 14A
9B88 FE 66 28 0F FE 6A 28 0B 136
9B90 FE 6E 28 07 10 E2 AF 035
9B98 02 02 3E 01 28 00 00 111
9BE0 00 3A 06 A7 CA 05 F5 3A 1E1
9BE8 16 A7 FE 05 38 06 F1 E6 105
9BF0 1F C0 18 04 F1 E6 03 00 1D
9BF8 06 03 21 36 07 28 0E 13F
9C00 00 A7 28 08 11 05 00 0D 1CA
9C08 19 10 F3 C9 2A 19 A7 7D 14C
9C10 C6 08 67 FD 21 A7 07 06 183
9C18 0B 3A 15 28 28 18 E5 0E 155
9C20 00 58 92 FE 65 28 15 15C
9C28 05 67 28 11 FE 6D 28 10 14C
9C30 FE 20 20 00 EC E1 28 15C
9C38 20 20 00 EC E1 28 21 15C
9C40 93 23 16 28 21 21 15C
9C48 E6 07 20 1 F3E 01 24 1D 13E
9C50 77 04 2C 2C 20 21 0D 103
9C58 74 02 0D 36 00 01 E1 C9 134
9C60 00 00 00 00 00 00 00 00 13E
9C68 18 05 0E 05 CD 92 FE 1C5
9C70 67 28 15 FE 68 28 11 FE 144
9C78 6F 28 20 20 20 00 EC 117
9C80 E1 24 2D 7E 00 20 10C 19E
9C88 CD 29 91 E6 07 20 1 F3 1C3
9C90 02 18 8C 00 00 00 3A 06 116
9C98 A7 C6 02 E6 07 06 03 05 125
9CA0 0D 21 36 A7 0D 7E 00 A7 1DD
9CA8 28 09 0D E5 C5 CD BE 9C 1FD
9CC0 01 0D 66 02 CD 58 02 FE 1F8
9CC8 3C 28 0C FE 8D 28 0E 179
9CD0 BE 2A 04 FE 8F 20 03 CD 137
9CD8 27 94 2C 0D 75 01 70 FE 185
9CE0 1A 08 05 0D 30 00 00 C9 133
9CE8 0D 7E 04 FE 01 20 18 0E 133
9CF0 98 92 FE 28 05 FE 7A 10A
9CF8 C2 FF 90 2D 36 00 00 3E 1AF
9D00 01 32 1F 28 07 CD 58 92 179
9D08 78 78 8F FE 78 28 0E 117
9D10 FE 1F C4 9D 97 20 C4 109
9D18 A1 9D 91 FE 21 CA 91 9D 1E3
9D20 22 CA 90 FE 16 24 28 79 FE
9D28 FE 23 28 7F FE 08 28 79 169
9D30 FE 0E 28 7F FE 08 28 79 169
9D38 FE 0D 28 71 FE 0C 28 71 14F
9D40 FE 10 28 71 FE 16 28 69 140
9D48 FE 1E 28 69 FE 13 28 69 144
9D50 FE 15 28 65 FE 1D 28 61 144
9D58 FE 14 28 61 FE 1C 28 60 13A
9D60 FE 01 28 60 FE 0E 28 59 112
9D68 FE 19 28 55 FE 0E 28 55 111
9D70 FE 07 28 51 FE 18 28 4D 10C
9D78 FE 03 28 4D FE 1A 28 49 10C
9D80 FE 04 28 49 FE 0E 28 49 10C
9D88 FE 17 28 47 FE 00 C0 1D 1F
9D90 7E 03 FE 01 CA 05 93 C3 1A5
9D98 E5 92 00 00 00 00 98 18 105
9DA0 2A 3E 10 00 00 3E 18 10 14A
9DA8 22 3E 08 18 16 18 10 18F
9DB0 1A 3E 08 18 16 3E 11 18 18F
9DB8 12 3E 12 18 0E 3E 13 18 1F1
9DC0 CA 3E 02 18 06 3E 06 18 1C4

90CB 02 3E 04 CD 04 94 DD 36 1C2
90DD 00 00 C9 CD 27 94 18 F6 15F
90DE 3E 18 1B 07 FE 1A 18 EB 1BB
90DF 3E 07 18 07 FE 3E 15 EB 1B2
90EB 3E 03 18 DF 37 18 DB 1B0
90FD 3E 19 1B 07 FE 3E 15 EB 1B2
90FB 3E 01 18 CF 00 00 00 FE 124
90FO 1F 2B F5 FE 20 2D ED FE 1A0
90GH 22 2B EF FE 24 2B E5 FE 1A0
90IO 21 2B DD FE 23 2B E5 FE 1A2
90IB 0B 2B A6 FE 12 2B A2 FE 1A0
90IC 20 2B A2 FE 11 2B 9E FE 1A6
90ID 02 2B 9E FE 13 2B 9A FE 199
90IE 0B 2B 9E FE 06 2B 9C FE 1B8
90IF 0B 2B 9E FE 12 2B 9E FE 1A7
90IG 2B 2B 9E FE 18 2B 8C FE 198
90IH 1D 2B 8B FE 1E 2B 89 FE 198
90II 19 2B 89 FE 1C 2B 85 FE 1B8
90IJ 0E 2B 85 FE 1D 2B 81 FE 176
90IK 07 CA 94 9D FE 0A E4 10B
90IL 0B 9E FE 15 CA 94 FE 01 FA6
90IM 0A E8 9D FE 0C E4 9D 1A8
90IN 0E 9D FE 0C E4 9D 1A8
90IO 0E 9D FE 0C E4 9D 1A8
90IP 0E 01 CA F5 C3 D5 92 17A
90IQ 00 00 00 3A 06 AF E6 03 1A8
90IR 0B 3A 16 AF FE 0B D8 2A 1B8
90IS 2E AF 23 7C FE 01 20 05 19E
90IT 7D FE 0A 2B 2F 2B AF 17E
90IU 00 21 00 22 2E AF CD 1AE
90IV 01 E6 01 2B 06 AF 21 19F
90IW 02 00 18 05 3E 01 21 02 1B1
90IX 1C 32 2A AF 22 2B 9E 1E1
90IY 01 32 2F AF FE 02 32 10 1A5
90IZ 07 C3 9B 95 00 00 3A CF 1E2
90JA 2F AF FE 01 C0 3A 06 AF 176
90JB 3C E6 07 C0 21 2B 00 01 133
90JC 05 00 CD 22 2F 2B AF 17D
90JD 3A 2A AF AF 2B 21 3A 30 1A5
90JE 07 FE 01 20 93 3E 02 32 1A5
90JF 0B 30 AF CD 9B 24 24 C3 1DA
90JG 27 94 7C AF 18 25 22 1A2
90JH 1B 2B AF 18 16 00 00 3A 13A
90JI 20 30 AF FE 01 2B 14 7C FE 1B8
90JJ 2B 1C 20 07 AF 32 2B AF C3 1B7
90JK 37 94 3E 01 22 30 A7 C3 1D6
90JL 07 93 3E 02 32 30 A7 CD 1B0
90JM 27 94 2A 22 2B AF C3 9E 1E2
90JN 4B 93 00 00 3A 34 AF FE 1A6
90JO 01 C0 3A 06 AF E6 03 CD 151
90JP 3A 35 AF FE 1E 17E
90JQ 32 35 AF C9 3E 28 32 35 1A4
90JR 07 2A 31 AF 36 00 23 36 1B8
90JS 00 25 36 00 AF 32 34 AF 115
90JT 09 00 00 00 00 00 00 00 00
90JU 2A 10 AF 06 05 C3 5F CD 1A3
90JV 0B 9E 93 11 00 A2 CD 3E 07 12C
90JW 01 E5 CD C9 93 11 29 A2 1CB
90JX 0B CD 3F 0F 21 10 E6 1A 9E
90JY 0A FF CD 1C 91 37 3A 3A 1A8
90JZ 0A 07 20 32 00 AF F5 C6 192
90KA 30 32 FA 12 D6 FF CD 1C 126
90KB 91 14 AF 2B 10 3A 56 AF 198
90KC 0A FE 01 2B 09 CD FB 94 C3 151
90KD 1D 90 00 00 00 21 C7 EB 17D
90KE 0E 05 E5 06 11 AF CD 85 1BB
90KF 01 E1 CD DD 91 00 20 F2 1CC
90KG 21 1F 01 11 4C A5 CD 11 15
90KH 9E 05 06 05 21 BF D1 E5 11 147
90KI 0B 85 A5 CD A1 91 E1 CD DD 1E7
90KJ 01 10 F3 21 47 01 11 CB 1EA
90KK 00 00 00 00 00 00 00 00 00
90KL 0F CD A6 90 21 39 02 17E
90KM 11 DE A5 CD A1 91 21 45 1F9
90KN 02 CD EA 91 CD F3 A0 CD 147
90KO 16 A1 CD 55 A0 CD 5C A0 142
90KP 02 73 A0 23 1A 90 00 1A0
90KQ 00 11 A2 CD 3F 0F 16 1A6
90KR 0F CD 1C 7D 91 CD 16 A1 2A 27
90KS 1C 17 7D FE 0B 2B 07 2C 1A6
90KT 22 1C A7 01 11 4C A5 CD 11 15
90KU 18 F6 00 00 21 00 00 10F
90KV 0E CB 18 05 00 21 00 D0 0E 1F2
90KW 19 E5 06 1E AF CD 85 91 1E4
90KX 01 CD DD 91 00 20 F2 91 104
90KY 00 00 00 21 1A 90 00 1A0
90KZ 0A CD A1 91 14 CD 06 1A 9A
90LA 07 21 AB 0D 11 09 A6 CD 2D
90LB 17 95 21 05 E5 11 B1 A3 11C
90LC 07 CD E7 A0 00 00 00 00 00
90LD 0A CD E7 A0 21 05 EA 11 117
90LE 0A E2 CD E7 A0 21 85 E5 136
90LF 11 BA A2 CD E7 A0 21 09 1BB
90LG 02 11 3F A6 CD A1 91 CD 17A
90LH 14 0F 21 3C DD 00 00 00 00
90LI 0C CD 15 95 CD 14 0F 21 14 9C
90LJ 0B 11 19 A3 CD E7 A0 21 136
90LK 01 D3 11 A5 A6 CD A1 91 14F
90LM 21 9C 03 11 DE A5 CD A1 192
90LN 01 CD 14 0F C3 E7 01 E5 1A1
90LO 0B CD CF 91 E1 23 C3 CF 91 154

A0FO 00 00 00 00 21 00 00 22 09 14C
A0FB AF 22 27 AF 3E 04 32 00 1E8
A0FC 10 07 3E 01 32 0C AF 21 03 1EF
A0FD 04 22 1C AF 3F 32 56 AF 1C7
A0FE 11 00 00 00 00 00 00 AF 32 1A7
A0FF 0B 07 32 0F AF 32 1F AF 1B7
A100 32 20 32 22 AF 32 2F 1A7
A101 07 32 34 AF 3E 3F 32 13 176
A102 37 3E 01 32 15 AF 32 17 110
A103 07 3E 01 32 15 AF 32 17 110
A104 32 35 AF 3E 32 37 16 AF 172
A105 02 29 91 E6 01 20 03 AF 140
A106 18 02 3E 01 32 14 AF AF 1F5
A107 32 36 32 32 37 AF 32 1A7
A108 37 3E 01 32 39 AF 3E 02 138
A109 32 3E AF 32 43 AF 21 00 154
A110 00 22 2E AF 06 0E 21 47 173
A111 07 36 01 23 10 FB 09 00 1D5
A112 00 00 00 00 2B 43 33 2B 1CC
A113 47 42 2B 43 52 32 2B 47 1ED
A114 01 52 2B 47 52 2B 47 52 10B
A115 2B 23 41 00 00 00 2B 41 108
A116 32 2B 46 31 52 32 2B 47 1A7
A117 46 32 2B 41 34 52 00 1C9
A118 00 00 2B 43 46 30 2B 47 136
A119 2B 23 47 00 00 00 2B 23 10F
A120 43 33 2B 44 31 2B 43 1B1
A121 43 24 2B 44 28 23 194
A122 24 2B 44 28 23 44 1B5
A123 24 2B 44 28 23 44 2B 19C
A124 24 2B 44 28 23 44 2B 19C
A125 24 2B 44 28 23 44 2B 19C
A126 24 2B 44 28 23 44 2B 19C
A127 24 2B 44 28 23 44 2B 19C
A128 24 2B 44 28 23 44 2B 19C
A129 24 2B 44 28 23 44 2B 19C
A130 24 2B 44 28 23 44 2B 19C
A131 24 2B 44 28 23 44 2B 19C
A132 24 2B 44 28 23 44 2B 19C
A133 24 2B 44 28 23 44 2B 19C
A134 24 2B 44 28 23 44 2B 19C
A135 24 2B 44 28 23 44 2B 19C
A136 24 2B 44 28 23 44 2B 19C
A137 24 2B 44 28 23 44 2B 19C
A138 24 2B 44 28 23 44 2B 19C
A139 24 2B 44 28 23 44 2B 19C
A140 24 2B 44 28 23 44 2B 19C
A141 24 2B 44 28 23 44 2B 19C
A142 24 2B 44 28 23 44 2B 19C
A143 24 2B 44 28 23 44 2B 19C
A144 24 2B 44 28 23 44 2B 19C
A145 24 2B 44 28 23 44 2B 19C
A146 24 2B 44 28 23 44 2B 19C
A147 24 2B 44 28 23 44 2B 19C
A148 24 2B 44 28 23 44 2B 19C
A149 24 2B 44 28 23 44 2B 19C
A150 24 2B 44 28 23 44 2B 19C
A151 24 2B 44 28 23 44 2B 19C
A152 24 2B 44 28 23 44 2B 19C
A153 24 2B 44 28 23 44 2B 19C
A154 24 2B 44 28 23 44 2B 19C
A155 24 2B 44 28 23 44 2B 19C
A156 24 2B 44 28 23 44 2B 19C
A157 24 2B 44 28 23 44 2B 19C
A158 24 2B 44 28 23 44 2B 19C
A159 24 2B 44 28 23 44 2B 19C
A160 24 2B 44 28 23 44 2B 19C
A161 24 2B 44 28 23 44 2B 19C
A162 24 2B 44 28 23 44 2B 19C
A163 24 2B 44 28 23 44 2B 19C
A164 24 2B 44 28 23 44 2B 19C
A165 24 2B 44 28 23 44 2B 19C
A166 24 2B 44 28 23 44 2B 19C
A167 24 2B 44 28 23 44 2B 19C
A168 24 2B 44 28 23 44 2B 19C
A169 24 2B 44 28 23 44 2B 19C
A170 24 2B 44 28 23 44 2B 19C
A171 24 2B 44 28 23 44 2B 19C
A172 24 2B 44 28 23 44 2B 19C
A173 24 2B 44 28 23 44 2B 19C
A174 24 2B 44 28 23 44 2B 19C
A175 24 2B 44 28 23 44 2B 19C
A176 24 2B 44 28 23 44 2B 19C
A177 24 2B 44 28 23 44 2B 19C
A178 24 2B 44 28 23 44 2B 19C
A179 24 2B 44 28 23 44 2B 19C
A180 24 2B 44 28 23 44 2B 19C
A181 24 2B 44 28 23 44 2B 19C
A182 24 2B 44 28 23 44 2B 19C
A183 24 2B 44 28 23 44 2B 19C
A184 24 2B 44 28 23 44 2B 19C
A185 24 2B 44 28 23 44 2B 19C
A186 24 2B 44 28 23 44 2B 19C
A187 24 2B 44 28 23 44 2B 19C
A188 24 2B 44 28 23 44 2B 19C
A189 24 2B 44 28 23 44 2B 19C
A190 24 2B 44 28 23 44 2B 19C
A191 24 2B 44 28 23 44 2B 19C
A192 24 2B 44 28 23 44 2B 19C
A193 24 2B 44 28 23 44 2B 19C
A194 24 2B 44 28 23 44 2B 19C
A195 24 2B 44 28 23 44 2B 19C
A196 24 2B 44 28 23 44 2B 19C
A197 24 2B 44 28 23 44 2B 19C
A198 24 2B 44 28 23 44 2B 19C
A199 24 2B 44 28 23 44 2B 19C
A200 24 2B 44 28 23 44 2B 19C
A201 24 2B 44 28 23 44 2B 19C
A202 24 2B 44 28 23 44 2B 19C
A203 24 2B 44 28 23 44 2B 19C
A204 24 2B 44 28 23 44 2B 19C
A205 24 2B 44 28 23 44 2B 19C
A206 24 2B 44 28 23 44 2B 19C
A207 24 2B 44 28 23 44 2B 19C
A208 24 2B 44 28 23 44 2B 19C
A209 24 2B 44 28 23 44 2B 19C
A210 24 2B 44 28 23 44 2B 19C
A211 24 2B 44 28 23 44 2B 19C
A212 24 2B 44 28 23 44 2B 19C
A213 24 2B 44 28 23 44 2B 19C
A214 24 2B 44 28 23 44 2B 19C
A215 24 2B 44 28 23 44 2B 19C
A216 24 2B 44 28 23 44 2B 19C
A217 24 2B 44 28 23 44 2B 19C
A218 24 2B 44 28 23 44 2B 19C
A219 24 2B 44 28 23 44 2B 19C
A220 24 2B 44 28 23 44 2B 19C
A221 24 2B 44 28 23 44 2B 19C
A222 24 2B 44 28 23 44 2B 19C
A223 24 2B 44 28 23 44 2B 19C
A224 24 2B 44 28 23 44 2B 19C
A225 24 2B 44 28 23 44 2B 19C
A226 24 2B 44 28 23 44 2B 19C
A227 24 2B 44 28 23 44 2B 19C
A228 24 2B 44 28 23 44 2B 19C
A229 24 2B 44 28 23 44 2B 19C
A230 24 2B 44 28 23 44 2B 19C
A231 24 2B 44 28 23 44 2B 19C
A232 24 2B 44 28 23 44 2B 19C
A233 24 2B 44 28 23 44 2B 19C
A234 24 2B 44 28 23 44 2B 19C
A235 24 2B 44 28 23 44 2B 19C
A236 24 2B 44 28 23 44 2B 19C
A237 24 2B 44 28 23 44 2B 19C
A238 24 2B 44 28 23 44 2B 19C
A239 24 2B 44 28 23 44 2B 19C
A240 24 2B 44 28 23 44 2B 19C
A241 24 2B 44 28 23 44 2B 19C
A242 24 2B 44 28 23 44 2B 19C
A243 24 2B 44 28 23 44 2B 19C
A244 24 2B 44 28 23 44 2B 19C
A245 24 2B 44 28 23 44 2B 19C
A246 24 2B 44 28 23 44 2B 19C
A247 24 2B 44 28 23 44 2B 19C
A248 24 2B 44 28 23 44 2B 19C
A249 24 2B 44 28 23 44 2B 19C
A250 24 2B 44 28 23 44 2B 19C
A251 24 2B 44 28 23 44 2B 19C
A252 24 2B 44 28 23 44 2B 19C
A253 24 2B 44 28 23 44 2B 19C
A254 24 2B 44 28 23 44 2B 19C
A255 24 2B 44 28 23 44 2B 19C
A256 24 2B 44 28 23 44 2B 19C
A257 24 2B 44 28 23 44 2B 19C
A258 24 2B 44 28 23 44 2B 19C
A259 24 2B 44 28 23 44 2B 19C
A260 24 2B 44 28 23 44 2B 19C
A261 24 2B 44 28 23 44 2B 19C
A262 24 2B 44 28 23 44 2B 19C
A263 24 2B 44 28 23 44 2B 19C
A264 24 2B 44 28 23 44 2B 19C
A265 24 2B 44 28 23 44 2B 19C
A266 24 2B 44 28 23 44 2B 19C
A267 24 2B 44 28 23 44 2B 19C
A268 24 2B 44 28 23 44 2B 19C
A269 24 2B 44 28 23 44 2B 19C
A270 24 2B 44 28 23 44 2B 19C
A271 24 2B 44 28 23 44 2B 19C
A272 24 2B 44 28 23 44 2B 19C
A273 24 2B 44 28 23 44 2B 19C
A274 24 2B 44 28 23 44 2B 19C
A275 24 2B 44 28 23 44 2B 19C
A276 24 2B 44 28 23 44 2B 19C
A277 24 2B 44 28 23 44 2B 19C
A278 24 2B 44 28 23 44 2B 19C
A279 24 2B 44 28 23 44 2B 19C
A280 24 2B 44 28 23 44 2B 19C
A281 24 2B 44 28 23 44 2B 19C
A282 24 2B 44 28 23 44 2B 19C
A283 24 2B 44 28 23 44 2B 19C
A284 24 2B 44 28 23 44 2B 19C
A285 24 2B 44 28 23 44 2B 19C
A286 24 2B 44 28 23 44 2B 19C
A287 24 2B 44 28 23 44 2B 19C
A288 24 2B 44 28 23 44 2B 19C
A289 24 2B 44 28 23 44 2B 19C
A290 24 2B 44 28 23 44 2B 19C
A291 24 2B 44 28 23 44 2B 19C
A292 24 2B 44 28 23 44 2B 19C
A293 24 2B 44 28 23 44 2B 19C
A294 24 2B 44 28 23 44 2B 19C
A295 24 2B 44 28 23 44 2B 19C
A296 24 2B 44 28 23 44 2B 19C
A297 24 2B 44 28 23 44 2B 19C
A298 24 2B 44 28 23 44 2B 19C
A299 24 2B 44 28 23 44 2B 19C
A300 24 2B 44 28 23 44 2B 19C
A301 24 2B 44 28 23 44 2B 19C
A302 24 2B 44 28 23 44 2B 19C
A303 24 2B 44 28 23 44 2B 19C
A304 24 2B 44 28 23 44 2B 19C
A305 24 2B 44 28 23 44 2B 19C
A306 24 2B 44 28 23 44 2B 19C
A307 24 2B 44 28 23 44 2B 19C
A308 24 2B 44 28 23 44 2B 19C
A309 24 2B 44 28 23 44 2B 19C
A310 24 2B 44 28 23 44 2B 19C
A311 24 2B 44 28 23 44 2B 19C
A312 24 2B 44 28 23 44 2B 19C
A313 24 2B 44 28 23 44 2B 19C
A314 24 2B 44 28 23 44 2B 19C
A315 24 2B 44 28 23 44 2B 19C
A316 24 2B 44 28 23 44 2B 19C
A317 24 2B 44 28 23 44 2B 19C
A318 24 2B 44 28 23 44 2B 19C
A319 24 2B 44 28 23 44 2B 19C
A320 24 2B 44 28 23 44 2B 19C
A321 24 2B 44 28 23 44 2B 19C
A322 24 2B 44 28 23 44 2B 19C
A323 24 2B 44 28 23 44 2B 19C
A324 24 2B 44 28 23 44 2B 19C
A325 24 2B 44 28 23 44 2B 19C
A326 24 2B 44 28 23 44 2B 19C
A327 24 2B 44 28 23 44 2B 19C
A328 24 2B 44 28 23 44 2B 19C
A329 24 2B 44 28 23 44 2B 19C
A330 24 2B 44 28 23 44 2B 19C
A331 24 2B 44 28 23 44 2B 19C
A332 24 2B 44 28 23 44 2B 19C
A333 24 2B 44 28 23 44 2B 19C
A334 24 2B 44 28 23 44 2B 19C
A335 24 2B 44 28 23 44 2B 19C
A336 24 2B 44 28 23 44 2B 19C
A337 24 2B 44 28 23 44 2B 19C
A338 24 2B 44 28 23 44 2B 19C
A339 24 2B 44 28 23 44 2B 19C
A340 24 2B 44 28 23 44 2B 19C
A341 24 2B 44 28 23 44 2B 19C
A342 24 2B 44 28 23 44 2B 19C
A343 24 2B 44 28 23 44 2B 19C
A344 24 2B 44 28 23 44 2B 19C
A345 24 2B 44 28 23 44 2B 19C
A346 24 2B 44 28 23 44 2B 19C
A347 24 2B 44 28 23 44 2B 19C
A348 24 2B 44 28 23 44 2B 19C
A349 24 2B 44 28 23 44 2B 19C
A350 24 2B 44 28 23 44 2B 19C
A351 24 2B 44 28 23 44 2B 19C
A352 24 2B 44 28 23 44 2B 19C
A353 24 2B 44 28 23 44 2B 19C
A354 24 2B 44 28 23 44 2B 19C
A355 24 2B 44 28 23 44 2B 19C
A356 24 2B 44 28 23 44 2B 19C
A357 24 2B 44 28 23 44 2B 19C
A358 24 2B 44 28 23 44 2B 19C
A359 24 2B 44 28 23 44 2B 19C
A360 24 2B 44 28 23 44 2B 19C
A361 24 2B 44 28 23 44 2B 19C
A362 24 2B 44 28 23 44 2B 19C
A363 24 2B 44 28 23 44 2B 19C
A364 24 2B 44 28 23 44 2B 19C
A365 24 2B 44 28 23 44 2B 19C
A366 24 2B 44 28 23 44 2B 19C
A367 24 2B 44 28 23 44 2B 19C
A368 24 2B 44 28 23 44 2B 19C
A369 24 2B 44 28 23 44 2B 19C
A370 24 2B 44 28 23 44 2B 19C
A371 24 2B 44 28 23 44 2B 19C
A372 24 2B 44 28 23 44 2B 19C
A373 24 2B 44 28 23 44 2B 19C
A374 24 2B 44 28 23 44 2B 19C
A375 24 2B 44 28 23 44 2B 19C
A376 24 2B 44 28 23 44 2B 19C
A377 24 2B 44 28 23 44 2B 19C
A378 24 2B 44 28 23 44 2B 19C
A379 24 2B 44 28 23 44 2B 19C
A380 24 2B 44 28 23 44 2B 19C
A381 24 2B 44 28 23 44 2B 19C
A382 24 2B 44 28 23 44 2B 19C
A383 24 2B 44 28 23 44 2B 19C
A384 24 2B 44 28 23 44 2B 19C
A385 24 2B 44 28 23 44 2B 19C
A386 24 2B 44 28 23 44 2B 19C
A387 24 2B 44 28 23 44 2B 19C
A388 24 2B 44 28 23 44 2B 19C
A389 24 2B 44 28 23 44 2B 19C
A390 24 2B 44 28 23 44 2B 19C
A391 24 2B 44 28 23 44 2B 19C
A392 24 2B 44 28 23 44 2B 19C
A393 24 2B 44 28 23 44 2B 19C
A394 24 2B 44 28 23 44 2B 19C
A395 24 2B 44 28 23 44 2B 19C
A396 24 2B 44 28 23 44 2B 19C
A3



PART 2

■ルリタテハ

(注) このゲームにはグラフィックV-RAMが必要です。

「HEAD-ON」はインベーダーの次くらいにでてきたゲームで、やったことのある人も多いと思います。あまりバツとしませんでした。なかなか面白いゲームでした。原型のままに作ってもつまらないので、コースの一部を屈曲させ、さらに壁は5箇所が一定時間ごとに開閉するようにして少し改良してみました。名付けてHEAD-ON part2です。

1 遊び方

モニタのJコマンドを使って9000番地からスタートさせてください。まずタイトルと説明が表示されます。車はテンキーの[2]、[4]、[6]、[8]で車線を変えます。このキーの押す方向はTV画面に対してで、車の進行方向とは関係ありません。

要するに、車線変更したい側のキーを押せばいいのです。当然壁のあるところでは車線は変えられず、壁のとぎれていところのみ車線を変えられます。5箇所の壁は一定時間で開閉します。閉じているときは車線を変えることができないので、速度をうまく調節してタイミングをとってください。

車線は一度に2車線まで変えることができます。

スベス キーで車の速度を上げられます。加速にはけっこう時間がかかり、減速はかなり速くなるようにしてあります。自分の車のスピードを上げるとしばらくして敵もスピードアップしてきます。その間は自分の車の方が速いので有利になります。減速のときは逆に敵はしばらくしてから減速するので、その間は敵の方が速いということになります。

自分の車は左回り、敵の車は右回りと決まっているので、常に正面衝突の危険があるわけです。敵はこちらの動きに合わせて車線を変えてきます。敵の裏をかくように操作すればいいのですが、敵は乱数で動きを決めているので、当然車線を変えるはずが、変えなかったり、ときには敵も2車線分動いたりして、裏をかかれることの方が多いみたいです。

第1面の得点はドット10点、タイヤ60点、ボーナス500点となっています。ドットをすべて消すと一面終了でボーナスが付きます。次の面からはドットが10点、タイヤが20

点、ボーナスが100点ずつプラスされます。ドット数が20個以下になると敵はドットをタイヤに変えます。タイヤの方が点が高いので、これをうまく利用すると高得点が期待できます。第8面からは敵は2台となり、より難しくなります。

5,000点を越えると車は1台追加されます。車がすべて衝突しなくなるとゲーム・オーバーです。20面でギアアップになっていますが、ここまでいくことはないと思います。車が衝突したときはまた最初のドットが全部ある状態から始まるので、一面消すのはなかなかつらくなっています。ハイ・スコアを0にしたいときは、A5C2、A5C3番地の値を00にしてください。

2 プログラムの入力

モニタのMコマンドを使って9000番地からリストどおりに打ち込んでください。このMコマンドはいちいち[CR]を押さなくてもいいので、これだけの量のリストとなるとちょっと大変だと思います。'81年6月号のセルフ・リロケータブル・デバッグを持っている方はこのWコマンドを使って入力の方がかなり早くできると思います。

入力し終わったら、まず5コマンドでテープに録っておきましょう。それからチェック・サムを調べて、すべてOKでしたらもう一度モニタの5コマンドでテープに入れます。ファイル・ネームは好きな名前にしてください。

S-ADR、\$に9000、E-ADR、\$にA5BB、J-ADR、\$に9000と打ち込んでテープを作ると、次回にLOADしたときにはオート・スタートになります。

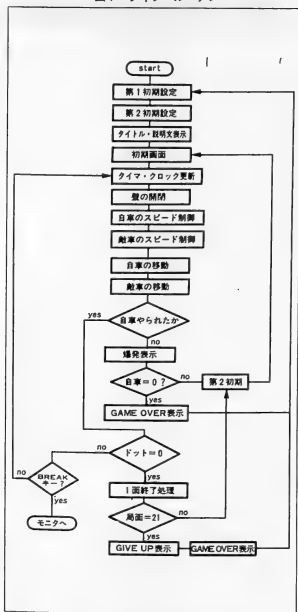
3 プログラムの解説

プログラムはオール・マシン語です。各処理はすべてサブルーチン化して、メイン・ルーチンでは図1のフローチャートに示すように各サブルーチンを順に呼び出していく形をとっています。

そして、このサブルーチンの呼び出すタイミングを決めるのがメイン・クロックの値で、この数値によって各サブルーチンへ行った行きなかつたりして速度に差をつけます。

メイン・ループをこのままで使くと、あまりの早さに手

図1 メイン・ルーチン



がつけられないので、タイマを入れて時間つぶしをして遅くしています。それでも高速のときは少し速すぎるくらいです。

メイン・ルーチン内に **BREAK** キーを押したときの判定をつけてあるので、ゲーム中に **BREAK** キーを押すとモニタに炸れます。ゲーム・オーバー時や、タイトル表示時には判定をしていないので、一旦、ゲームをスタートさせてから **BREAK** キーを押してください。リセット・スイッチを押してもいいのですが、このスイッチは「禁断のスイッチ」とも呼ばれるもので、さわるのは恐ろしいので

各マシン語ルーチンの内容については表1に示します。細かいサブルーチンのフローチャートはとても書き切れないので省略しました。ワーク・エリアの内容を表2に、データ・エリアの内容を表3に示します。

このゲームのようにドット・グラフィックを使うときはドット・グラフィック画面上で敵と衝突したとか、ドットがあるかなどという判定をするのは大変に面倒なことにな

表1 マシン語ルーチンの内容

アドレス	内 容
9000	メイン・ルーチン
903D	ビデオ RAM のスイッチを ON にする。
904B	キー入力のためのサブルーチン
9058	1 キャラクタ分のドットをクリア
9069	キャラクタ画面クリア
9078	ドット画面クリア
9087	ハイ・スコア判定
90AB	10 進出力サブルーチン
9102	タイマ
9111	乱数を作るサブルーチン
9135	クロック類 (メイン・自車・敵車) の更新
915E	車を追加するかの判定
9195	各種プリント・サブルーチン群
91DE	画面に壁を描く
92C6	自車、敵車、ダイアを描くサブルーチン
931B	車の初期と画面への表示
93FD	X・Y 値よりアドレス計算
942C	キャラクタ画面へ得点、車の数を描く。
9482	壁の閉閉
956F	自車のハイ・スピード処理
959B	敵車の "
95C7	自車移動のメイン・ルーチン
95EC	自車右へ移動
96A1	" 左へ "
971E	" 上へ "
9785	" 下へ "
97FC	自車の進行方向にあるものの判定
985C	敵車移動のメイン・ルーチン
988C	" " のサブルーチン
98C8	敵車右へ移動
994A	" 左へ "
99B3	" 上へ "
9A1C	" 下へ "
9AA8	敵車の進行方向にあるものの判定
9AF1	ゲームスタート時の局面数とドット、ダイア、ボーナス得点の表示
9BA0	一面終了時の処理
9C30	[CR] キー入力待ち
9C4F	爆発マークを描く
9CBC	タイトルと説明文の表示
9DC3	第 1 初期設定
9DE7	第 2 "

ります（ブロックくずしではやりましたが）。

そこで、ワーク・エリア内に40×25の仮想画面を作り、車の移動その他すべてをここで行なっています。そして、この車の動きに対応させてドット画面上の車を移動させています。そのため、車の動きは8ドットごとになるってしまい、ちょっとなめらかさに欠ける感じはしますが、仮想画面とCRT上の画面の両方を操作するので、プログラムは多少複雑になっています。

4 ゲームの変更点

- ①全体のスピード
9103番地の値(05)を小さくすると遅くなります。

表2 ワーク・エリア

アドレス	内 容
A5BD, A5BE	乱数値
A5BF	メイン・クロック値
A5C0, A5C1	スコア
A5C2, A5C3	ハイ・スコア
A5C4	車を追加したかのフラグ
A5C5	自車数
A5C6	局面数
A5C7	ドットの得点
A5C9	タイアの "
A5CB	ボーナスの "
A5CD, A5CE	自車のアドレス
A5CF, A5D0	" の方向値
A5D1	" の車線値
A5D3	" スピードの最高値
A5D5	自車クロック
A5D7~A5E6	敵車のアドレス, 方向値, 車線値で2 百分
A5E7	ドットの数
A5E9	敵車クロック
A5EB	敵車スピードの最高値
A5ED	敵車の待機クロック
A5EF	敵車数
A5F0	壁を開くか閉じるかのフラグ
A5F1	壁を開いている時間
A5F3	壁を閉じている時間
A5F5	衝突したときのフラグ
A5F6, A5F7	アドレス計算のときの先頭アドレスが 入る
A5FA~A5E3	ワーク・エリア上の仮想画面

表3 データ・エリア

アドレス	内 容
9DFC	CRT初期画面のデータ
A1EA	MUSICのデータ
A29B	壁のデータ
A2B9	ドットのデータ
A2C3	壁の角の部分のデータ
A2FF	自分の車のデータ(上, 下, 左, 右の順)
A327	タイアのデータ
A331	敵の車のデータ
A345	開閉する壁のデータ
A359	メッセージの文字群
A3A7	衝突時の爆発マークのデータ
A43A	タイトルの文字
A4E7	説明文の文字データ群

そこで、カーソル移動キーを使ってカーソルを動かし、ドットとして表示したい部分を"1"にします。ドットとして表示しない部分は1以外なら何でもいいのですが、テンキーには"1"とピリオドが近くにあります。ピリオドにしておいた方が後で見やすいでしょう。

8×8のマトリックスに"1"を書き終わったら、カーソルをマトリックス最下行より下の行に移動させてRUNさせてください。すると、それぞれの対応する行の右側に10進数でデータが表示されるので、紙などに書きとめておきましょう。プリンタのある人は自分でプリンタ出力の命令を付けてください。[CR] キーを押せば画面クリアされ、また新しいデータが作れます。

リスト1のプログラムでDATA文の内容をPOKEしてマシン語を一部使っていますが、そのアセンブル・リストをリスト2に示します。

MZ-80BではV-RAMはメイン・メモリと切り離されているので、ただV-RAM領域にPEEKやPOKEしてもダメなのです。メモリ後半をV-RAMにするスイッチをONにしてやらなくてはけません。このプログラムは7000番地から始まっていて、VRAMONというのはメモリ後半をV-RAMにオープンして、V-RAM領域へのPEEKやPOKEを可能にしています。VRAMOFでこの状態を元に戻しています。

《マシン語での実行例》

それではマシン語で図2のインベーターを描いてみましょう。まず、リスト1のドットデータ製作プログラムでデータを作ってみてください。図2右側に示すのと同じ値が表示されたと思います。このデータを使ってインベーターを描くプログラムをリスト3に示します。モニタの[M]コマンドで9000番地から入力し、[J]コマンドで9000番地からスタートさせてください。画面左上にインベーターが描かれます。マシン語を動かす前にドット画面をクリアしておくのを忘れないようにしましょう。

《BASICでの実行例》

BASICでドットを扱うのにPATTERN文というのがありますが、これのデータもリスト1のプログラムで作れそうなのですが、何と、ダメなのです。

5 おまけ

●●●簡易型ドット・データ製作プログラム

リスト1に示すBASICプログラムは簡易型の8×8ドット・データ製作プログラムです。ドット・グラフィックのデータは、立っているビットを調べて2進数から変換してもいいのですが、計算したり表を見たりするのは面倒なことです。そこでその手間を省くためにこのプログラムを作ってみました。

一番最初に動かすときはGOTO250[CR]としてください。RUNしてしまつたときは、[CR] キーを押せば大丈夫です。画面がクリアされ、8×8のマトリックスが表示されます。

```

80 LIMIT %6FFF
90 FOR A=28672 TO 28684: READ B: POKE A, B: NEXT
100 DIM B(7), D(7)
110 D1=53356
120 FOR C=0 TO 7
130 AD=D1+C*40-1
140 FOR J=0 TO 7: AD=AD+1: USR($7000): Y=PEEK(AD): USR($7007)
150 IF Y=49 THEN B(J)=1: GOTO 170
160 B(J)=0
170 NEXT
180 D5=B(0)+B(1)*2+B(2)*4+B(3)*8+B(4)*16+B(5)*32+B(6)*64+B(7)*128
190 D(C)=D5
200 NEXT
210 FOR I=0 TO 7: CURSOR 20, 2+I: PRINT D(I): NEXT
220 CURSOR 5, 20: PRINT "CRT CLEAR = CR key";
230 GET V$: IF V$="" THEN 230
250 GOSUB 1000
260 END
1000 CONSOLE C40: PRINT CHR$(6)
1010 FOR J=20 TO 9: FOR K=8 TO 15
1020 CURSOR K, J: PRINT "-": NEXT K, J
1030 CURSOR 0, 10: RETURN
1050 DATA 219, 232, 203, 255, 211, 232, 201, 219, 232, 203, 191, 24, 247

```

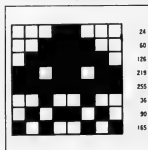
リスト2 V-RAMオープン&クローズ

```

; ORG 7000H
;
7000 DBE8 ; VRAMON: IN A, (0EH)
7002 CBF7 ; SET 7, A
7004 D3E8 ; VRAM1: OUT (0EH), A
7006 C9 ; RET
;
7007 DBE8 ; VRAMOF: IN A, (0EH)
7009 CBF7 ; RES 7, A
700B 1BF7 ; JR VRAM1
;
700D ; END

```

図2



マシン語のときはビットを「右←左」へ読み込んでいくのに、BASICでは逆に「左→右」へ読み込むようになっています。そこで、行番号110～140までをリスト4のように変えてください。これでPATTERN文に変えるデータが得られます。詳しいことはBASICマニュアルのp.59を読んでください。

それではBASICで図2に示したインペーダーを描いてみましょう。左右対象なので、リスト4で変更したプログラムでもまったく同じデータが表示されるはずですが、このデータを使ってインペーダーを描くプログラムをリスト5に示します。PATTERN文の積み重ね段数に“-”を付けると上から下へ描かれ、“-”を付けずに下から上へ描かれます。

リスト3 インペーダー表示プログラム

```

; ** DEMO PROGRAM **
;
; ORG 9000H
;
9000 CD2290 ; CALL VRAMON
9003 214BE6 ; LD HL, 0E64BH
9006 112D90 ; LD DE, DATA
9009 CD1490 ; CALL PRNT
900C CD3202 ; CALL 0562
900F CA0000 ; JP Z, 0
9012 1BF8 ; JR DEM1
;
9014 0608 ; LD B, B
9016 1A ; LD A, (DE)
9017 77 ; LD (HL), A
9018 05 ; PUSH DE
9019 112B00 ; LD DE, 40
901C 19 ; ADD HL, DE
901D D1 ; POP DE
901E 13 ; INC DE
901F 10F5 ; DJNZ PR1
9021 C9 ; RET
;
9022 DBE8 ; VRAMON: IN A, (0EH)
9024 CBF7 ; SET 7, A
9026 D3E8 ; OUT (0EH), A
9028 3E02 ; LD A, 2
902A D3F4 ; OUT (0F4H), A
902C C9 ; RET
;
902D 1B3C7EDB ; DATA: DB 24, 60, 126, 219, 255, 36, 90, 165
9031 FF245AAS ;
;
9035 ; END

```

6 おわりに

「おまけ」がずいぶん沢山になってしまいましたが、けっこう役に立つと思います。しかし、かなり簡易型なのでそのうちマシン語でちゃんとしたものを作ってみようと思います。

久しぶりにHEAD-ONをやったらなかなか面白くて、熱中してしまいました。壁の開閉まで付けたので少し難しくなりましたが、筆者はまだ高速のときには一面全部消したことがありません。低速だと何とかなるのですが、

リスト4 BASIC用への変換点

```

110 D1=53343
120 FOR C=7700 STEP-1
130 AD=D1+C*40+1
140 FOR J=0707:AD=AD-1:USR(¥7000):Y=PEEK(AD):USR(¥7007)

```

リスト5

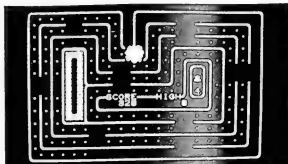
```

80 PRINTCHR$(6)
100 GRAPH 11,0,C
120 POSITION 64,48
140 A=CHR$(24)+CHR$(60)+CHR$(126)+CHR$(219)+CHR$(255)+CHR$(36)+CHR$(90)+CHR$(16)
150
160 PATTERN -B,A$
180 POSITION 80,55
200 PATTERN B,A$
220 END

```

参考にしたもの

- 馬場隆信: "HEAD-ON", I/O, '78年10月号
- ゲーム: "HEAD-ON"



マシン語 ダンプ・リスト

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9000	CD	C3	9D	CD	E7	9D	CD	8C	9C	CD	61	94	CD	35	91	CD	1C5
9001	02	91	CD	B2	94	CD	6F	95	CD	9B	95	CD	C7	95	CD	5C	196
9002	98	3A	F5	A5	FE	01	CA	4F	9C	3A	E7	A5	47	CA	0A	9B	192
9003	CD	A2	05	CA	00	00	1B	D4	00	00	00	00	00	DB	EB	CB	178
9004	FF	D3	EB	3E	02	D3	F4	C9	00	00	00	00	00	DB	EB	CB	178
9005	D3	EB	DB	EA	C9	00	00	00	00	00	00	00	00	00	00	00	104
9006	CD	C1	91	C1	C9	00	00	00	00	00	00	00	00	00	00	00	1E9
9007	91	00	20	FA	C9	00	00	00	00	00	00	00	00	00	00	00	118
9008	00	20	FA	C9	00	00	00	00	00	00	00	00	00	00	00	00	15F
9009	03	00	18	09	3A	CD	A5	47	3A	C2	A5	BB	00	2A	CD	A5	192
900A	22	C2	A5	11	6E	D2	18	03	00	00	00	00	00	00	00	00	131
900B	D9	90	01	EB	03	CD	D9	90	01	64	00	00	00	00	00	00	158
900C	00	CD	D9	90	7D	CD	30	12	13	3E	30	12	06	05	E1	7E	188
900D	FE	30	C0	36	00	23	10	F7	C9	C5	A7	3E	30	12	06	05	158
900E	03	3C	18	F9	C1	09	12	13	C9	00	00	11	05	D1	26	115	
900F	00	AF	01	0A	00	CD	D9	90	7D	CD	30	12	C9	00	00	00	1FE
Sum	70	63	42	35	D7	25	D3	FD	29	1C	B6	D0	25	B2	08	FA	1A7

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9100	00	00	25	10	06	1E	00	1B	7A	B3	20	F8	C9	00	00	00	165
9101	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	123
9102	19	22	BD	A5	7C	D1	E1	C9	CD	11	91	BB	DB	9B	18	FB	13E
9103	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	17F
9104	32	D5	A5	18	05	3A	D3	A5	18	F6	3A	E9	A5	47	28	05	125
9105	30	32	E9	A5	C9	3A	EB	A5	18	F7	00	00	00	00	00	00	189
9106	A5	3A	C4	A5	A7	CD	7E	01	20	07	A7	7D	FE	F4	DB	147	
9107	18	04	A7	FE	01	0B	7A	C5	A5	32	32	C5	A5	C6	30	32	13E
9108	25	D2	AF	32	C4	A5	3E	07	CD	50	0E	11	E4	A1	CD	3F	159
9109	0F	C9	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
910A	06	04	1A	77	23	13	10	FE	00	06	2B	AF	77	23	10	FC	12D
910B	C9	21	00	F3	1A	A7	CB	77	13	09	18	F8	00	00	00	00	10F
910C	C5	77	19	10	FC	C9	06	0B	1A	77	CD	D4	91	13	10	FB	146
910D	C9	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	195
910E	A5	21	FC	9D	01	E9	03	ED	B0	11	FA	A5	21	00	E0	06	1A0
910F	2B	0E	19	C5	E5	D3	32	92	D1	E1	13	25	C1	00	F3	10C	
Sum	A3	B8	9B	42	B5	C5	95	C2	D8	FD	DB	8A	F6	F7	1B	2B	171

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9200	00	28	04	D5	11	18	01	19	D1	06	28	18	E6	21	08	EF	16F
9201	CD	B0	92	21	BB	F6	CD	B8	92	21	BF	F1	06	05	CD	5B	19C
9202	90	23	10	FA	21	97	F1	06	04	CD	5B	90	23	10	FA	C9	118
9203	00	00	00	1A	FE	95	20	0B	E5	D5	11	9B	A2	CD	C6	91	104
9204	D1	E1	C9	FE	96	20	0B	E5	D5	11	A5	A2	CD	C6	91	D1	141
9205	E1	C9	FE	A5	20	0B	E5	D5	11	B9	A2	CD	C6	91	D1	E1	174
9206	C9	FE	9C	20	0B	E5	D5	11	C3	A2	CD	C6	91	D1	E1	C9	150
9207	FE	9D	20	0B	E5	D5	11	D7	A2	CD	C6	91	D1	E1	C9	FE	19D
9208	9E	20	0B	E5	D5	11	D7	A2	CD	C6	91	D1	E1	C9	FE	9F	149
9209	20	0B	E5	D5	11	E1	A2	CD	C6	91	D1	E1	C9	FE	87	C2	15F
920A	5B	90	E5	D5	11	AF	A2	CD	C6	91	D1	E1	C9	00	00	00	1A3
920B	E5	D5	11	EB	A2	CD	C6	91	D1	E1	C9	E5	D5	11	F5	A2	159
920C	CD	C6	91	D1	E1	C9	E5	D5	11	FF	A2	CD	C6	91	D1	E1	1E1
920D	C9	E5	D5	11	09	A3	CD	C6	91	D1	E1	C9	E5	D5	11	13	18D
920E	A3	CD	C6	91	D1	E1	C9	E5	D5	11	10	A3	CD	C6	91	D1	1C2
920F	E1	C9	00	00	00	E5	D5	11	27	A3	CD	C6	91	D1	E1	C9	1DE
Sum	F8	11	41	C5	B5	BF	E6	DB	5F	4F	63	71	F7	E1	72	AB	1BB

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9300	E5	D5	11	31	A3	CD	C6	91	D1	E1	C9	E5	D5	11	3B	A3	1E7
9301	CD	C6	91	D1	E1	C9	00	00	00	00	00	00	00	00	00	00	183
9302	A5	21	18	0E	22	CD	A5	3E	01	32	D1	A5	3E	0B	32	D3	1B5
9303	A5	32	05	A5	CD	BA	93	DD	21	07	A5	21	1B	00	00	75	17C
9304	00	DD	74	01	21	FE	FF	DD	75	02	DD	74	03	01	DD	134	
9305	77	04	AF	DD	77	05	3E	0C	32	E9	A5	32	EB	A5	3E	0A	197
9306	32	ED	A5	CD	8C	93	3A	EF	A5	01	C8	11	08	00	DD	138	
9307	19	21	02	1C	DD	75	00	DD	74	01	21	02	00	DD	75	02	173
9308	DD	74	01	3E	01	DD	77	04	AF	DD	77	05	00	DD	75	00	1D1
9309	6A	01	DD	7E	02	FE	02	28	11	FE	28	00	E5	CD	0A	1EA	
930A	94	36	EB	E1	CD	FD	93	C3	00	93	E5	CD	0A	94	36	EB	1B7
930B	E1	CD	FD	93	C3	0B	93	00	00	2A	CD	A5	3A	CF	A5	1E9	
930C	FE	02	20	0B	3E	93	CD	EF	93	C3	92	FE	7E	20	0B	1A8	
930D	3E	92	CD	EF	93	C3	DC	92	FE	7B	20	0B	3E	91	CD	EF	179
930E	93	C3	D1	92	3E	90	CD	EF	93	C3	C6	92	00	00	E5	1DE	
930F	F5	CD	0A	94	F1	77	E1	C3	FD	C3	00	00	00	E5	21	00	102
Sum	3A	79	E6	C9	07	68	68	B3	94	D3	34	2F	01	B5	00	D3	1E2

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum	
9400	E0	22	F4	A5	E1	11	40	01	18	AF	E5	21	FA	A5	02	F4	180	
9410	A5	E1	11	28	05	7D	24	F6	A5	FE	01	28	04	05	15B			
9420	19	10	FD	C1	58	16	00	19	C9	00	00	2A	C0	A5	11	107		
9430	A6	D2	CD	AB	90	2A	C2	A5	11	6E	D2	CD	AB	90	3A	C5	129	
9440	A5	C6	70	32	25	02	D1	0D	CD	C6	92	21	3F	02	11	117		
9450	71	A3	CD	98	91	A7	02	11	7B	A3	CD	98	91	00	00			
9460	00	CD	30	90	CD	69	90	CD	78	90	CD	F1	9A	CD	69	90	153	
9470	CD	0E	31	FE	AF	32	E7	AS	CD	18	93	CD	2C	9A	C9	00	188	
9480	00	3A	BE	AF	E6	0F	C0	3A	F0	A5	A7	28	11	3A	F3	14F		
9490	A5	A7	28	05	3D	32	F5	A5	C9	AF	32	F0	A5	3E	03	132		
94A0	F3	A2	21	17	0C	CD	32	95	21	15	1A	CD	32	95	21	05	17A	
94B0	CD	32	95	21	0C	05	CD	F4	94	21	0A	21	18	35	3A	F6A		
94C0	F1	A5	A7	28	05	3D	32	F1	A5	C9	3E	01	32	F0	A5	3E	17C	
94D0	05	32	F1	A5	21	17	0C	CD	32	95	21	15	1A	CD	32	95	21	05
94E0	21	05	CD	32	95	21	05	CD	17	95	21	0A	21	18	35	3A	F6A	
94F0	2A	00	00	00	E5	CD	0A	94	3E	96	CD	0E	95	E1	CD	F0	1A5	

Sum C4 EE F1 D0 6B 6B 00 4A B5 68 7A 26 71 12 85 BE 121

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum
9500	93	06	03	C5	11	4F	A3	CD	98	91	C1	F0	F6	C9	00	00	121
9510	77	CD	93	06	03	0A	9A	FE	CD	0A	9A	FE	CD	0E	95	E1	1CC
9520	CD	93	06	03	0A	9A	FE	CD	0A	9A	FE	CD	0E	95	E1	1CC	
9530	00	00	E5	CD	0A	9A	FE	CD	0A	9A	FE	CD	0E	95	E1	1CC	
9540	03	E5	C5	11	4F	A3	CD	C6	91	C1	E1	23	10	C3	C9	06	1A1
9550	93	06	03	C5	11	4F	A3	CD	98	91	C1	F0	F6	C9	00	00	121
9560	FD	93	06	03	0A	9A	FE	CD	0A	9A	FE	CD	0E	95	E1	1CC	
9570	BF	A5	E6	F1	0C	0E	13	CD	AB	90	CD	AB	90	20	09	3A	17E
9580	A5	E6	F1	0C	0E	13	CD	AB	90	CD	AB	90	20	09	3A	17E	
9590	32	03	A5	C9	3E	08	F8	00	00	3A	BE	A5	E6	F1	0C	18F	
95A0	CD	0E	13	CD	AB	90	CD	0A	9A	FE	CD	0E	95	E1	1CC		
95B0	3D	05	1A	C9	3A	EB	A5	FE	0C	0A	02	32	EB	A5	FE	0C	18F
95C0	3E	0C	18	F8	00	00	00	3A	D5	FE	01	01	2A	CD	A5	1A9	
95D0	CD	3E	0C	18	F8	00	00	3A	D5	FE	01	01	2A	CD	A5	1A9	
95E0	96	FE	8B	CA	1E	97	C3	85	97	00	00	CD	74	96	CD	11E	
95F0	6C	96	FE	8B	CA	1E	97	C3	85	97	00	00	CD	74	96	CD	11E

Sum 7A 57 18 49 3E 49 76 93 BA 31 9F D1 3D 07 F3 AE 1D5

9600	96	28	10	48	00	CD	78	96	01	88	FF	ED	43	CF	A5	C3	1F0	
9610	1E	97	0E	11	CD	48	90	C8	57	28	19	0E	12	CD	48	90	1A7	
9620	CD	57	20	1F	CD	96	FD	21	49	96	CD	82	96	CD	6C	176		
9630	96	CD	57	20	1F	CD	96	FD	21	49	96	CD	82	96	CD	6C	176	
9640	96	18	EB	FD	21	24	16	EB	E5	CD	89	96	CD	96	18	EB	1A5	
9650	F2	CD	82	96	CD	77	96	18	E5	CD	00	00	E5	CD	0A	94	123	
9660	36	00	E1	E5	CD	FD	93	CD	58	90	E1	C9	2A	CD	A5	C0	121	
9670	0A	7E	09	D9	C9	21	05	CD	58	90	E1	C9	2A	CD	A5	C0	121	
9680	09	C9	D9	21	05	CD	A5	D9	C9	21	05	CD	A5	D9	C9	21	05	1CA
9690	D9	21	05	CD	A5	D9	C9	21	05	CD	A5	D9	C9	21	05	CD	A5	180
96A0	00	7B	96	CD	6C	96	FE	28	0F	CD	78	96	CD	6C	18F			
96B0	96	CD	74	96	FE	28	0F	CD	78	96	CD	6C	18F					
96C0	CD	6C	96	CD	6C	96	FE	28	0F	CD	78	96	CD	6C	18F			
96D0	A5	C3	1E	97	01	78	00	ED	A3	CF	A5	C3	85	97	0E	12	39	
96E0	CD	48	90	C8	57	28	19	0E	12	CD	48	90	C8	57	20	14	122	
96F0	21	15	97	CD	77	96	CD	38	96	FD	21	08	97	CD	90	173		
Sum	61	71	26	DB	B2	65	26	4C	3C	B2	07	50	A5	FC	DA	96	17C	

Sum 61 71 26 DB B2 A5 26 4C 3C B2 07 05 45 FC DA 46 1CF

9710	96	18	F1	CD	82	96	CD	90	96	C3	2E	96	CD	89	1CE	97	195	
9720	96	CD	6C	96	FE	28	0F	CD	78	96	CD	6C	96	CD	6C	96	139	
9730	96	FE	28	0F	CD	78	96	CD	82	96	01	FE	FF	ED	43	CF	18E	
9740	A5	C3	A1	96	01	78	00	ED	A3	CF	87	28	11	CB	77	20	134	
9750	1A	FD	21	72	97	CD	97	96	CD	74	96	C3	2E	96	FD	21	187	
9760	7A	97	CD	90	96	CD	78	96	C3	2E	96	FD	21	51	97	C3	132	
9770	2E	96	CD	78	96	CD	90	96	18	F1	CD	74	96	CD	97	96	16F	
9780	18	E9	00	00	CD	82	96	CD	89	1CE	97	96	CD	89	1CE	97	14C	
9790	82	96	CD	6C	96	FE	28	0F	CD	78	96	CD	6C	96	FE	28	0F	14C
97A0	96	CD	74	96	CD	6C	96	FE	28	0F	CD	78	96	CD	0A	01	FE	107
97B0	FF	ED	43	CF	A5	C3	A1	96	01	02	00	ED	43	CF	A5	C3	107	
97C0	1E	97	0E	11	CD	48	90	C8	57	28	19	0E	12	CD	48	90	1A7	
97D0	21	E9	97	CD	96	18	EB	FD	21	F1	97	CD	90	96	C3	19C		
97E0	5F	97	FD	21	CF	97	C3	2E	96	CD	74	96	CD	90	96	18	EB	
97F0	F1	CD	78	96	CD	97	96	18	E9	00	00	FE	A5	C0	20	28	18B	

Sum 44 AE 1A 9B BC B5 A4 3C 1A 72 A4 B3 F9 A2 D0 25 19B

9800	96	28	10	48	00	CD	78	96	01	88	FF	ED	43	CF	A5	C3	1F0	
9810	A5	CD	4F	0E	00	2A	C0	A5	11	20	55	CD	22	0F	3A	C7	101	
9820	90	CD	5E	91	21	E7	A5	35	CD	BA	93	FE	EA	10	10	3E	194	
9830	07	CD	00	0E	11	0A	CD	3F	0F	3A	C9	A5	18	D2	FE	194		
9840	EB	28	10	FE	EB	28	0E	00	00	00	00	00	00	00	00	00	1F4	
9850	01	CD	3A	ED	A5	C0	28	05	CD	32	ED	A5	C9	D0	21	D7	100	
9860	A5	02	00	05	C5	CD	8C	9B	C1	00	3A	FE	A5	E1	20	11		
9870	01	C9	11	08	00	D0	19	10	EA	C9	00	00	D0	6E	00	00	1C4	
9880	66	01	ED	0A	94	DD	7E	05	77	E1	CD	F0	93	CD	85	14E		
9890	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	14E	
98A0	B3	99	C3	1E	97	01	78	00	ED	A3	CF	A5	C3	85	97	0E	12	39
98B0	F5	92	C3	1E	97	01	78	00	ED	A3	CF	A5	C3	85	97	0E	12	39
98C0	F5	92	C3	1E	97	01	78	00	ED	A3	CF	A5	C3	85	97	0E	12	39
98D0	20	1E	D0	35	01	D0	3A	00	CD	90	9A	CD	35	00	FE	95	108	

98E0	28	07	D0	36	02	78	C3	1C	9A	DD	36	02	8B	C3	B3	99	1E1
98F0	28	07	D0	36	02	78	C3	1C	9A	DD	36	02	8B	C3	B3	99	1E1
Sum	67	06	6B	9B	92	89	2C	4F	3C	69	2F	D0	50	24	79	DB	10F

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum	
9900	E6	03	28	2D	3A	D1	A5	DD	BE	04	A8	28	12	3B	17	CD	127	
9910	00	CD	30	90	CD	69	90	CD	78	90	CD	F1	9A	CD	69	90	153	
9920	01	E6	F1	0C	0E	13	CD	AB	90	CD	AB	90	20	09	3A	17E		
9930	FE	F0	21	25	99	18	E1	04	34	00	00	34	04	18	F2	CD	164	
9940	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	129	
9950	FE	96	20	06	CD	34	01	C3	E9	98	DD	35	01	CD	90	9A	127	
9960	DD	34	01	FE	26	32	CD	11	91	E6	03	28	28	3A	D1	186		
9970	A5	DD	BE	04	28	12	3B	00	35	00	00	34	04	18	F2	112		
9980	AB	90	CD	AB	90	20	09	3A	00	35	04	FD	21	A0	99	C3	166	
9990	18	99	00	00	34	01	FE	18	F1	0C	0E	13	CD	AB	90	CD	158	
99A0	35	00	00	35	04	18	F1	0C	0E	13	CD	00	DD	35	04	18	E9	158
99B0	00	00	00	00	35	00	CD	90	9A	FE	25	06	DD	34	00	00	1E0	
99C0	C3	00	34	9A	DD	35	00	CD	90	9A	FE	24	30	FE	25	28	32	AAE
99D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	129	
99E0	14	DD	34	01	00	34	04	FD	21	11	9A	C3	18	99	CD	11	156	
99F0	F1	E6	F1	0C	0E	13	CD	35	01	CD	35	04	FD	21	09	9A	C3	170
Sum	E5	E2	66	7A	30	77	8B	78	1C	60	AC	40	93	6C	41	E3	179	

マシン語ダンプ・リスト

```
9DB0 D2 11 6C A5 CD 98 91 21 FA D2 11 7B A5 CD 98 91 1FE
9DB0 CD 14 0F 16 64 CD 04 91 21 74 D2 11 7E A5 CD 98 1ED
9DA0 91 CD 14 0F 21 90 D3 C3 36 9C 00 00 00 00 D9 21 194
9DB0 55 00 01 19 00 CD 22 0F D9 C9 21 00 00 18 F2 D3
9DB0 00 00 00 21 00 00 22 0C A5 3E 04 32 C5 A5 E 01 C5
9DD0 32 C4 A5 32 C6 A5 32 C7 A5 32 EF A5 E 06 72 C9 10B
9DE0 A5 3E 32 C8 A5 C9 AF C2 F0 A5 32 F5 A5 E 05 105
9DF0 32 F1 A5 3E 03 F2 F3 A5 39 00 00 00 9C 95 A5 162
```

Sum 2F 6B D5 9B 73 DO CA 5B 86 53 39 74 F0 26 66 D5 146

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
9E00 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E10 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E20 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E30 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 150
9E40 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 150
9E50 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E60 A5 9C 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E70 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9E80 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 150
9E90 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 150
9EA0 A5 9C 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9EB0 A5 9A 95 95 95 95 95 95 95 95 95 95 95 95 95 150
9EC0 A5 96 A5 96 00 96 20 96 20 96 20 A5 20 20 A5 150
9ED0 20 96 20 96 20 96 20 96 20 96 20 96 20 A5 20 149
9EE0 20 A5 20 A5 20 A5 20 96 20 96 20 96 20 96 A5 142
9EF0 A5 96 A5 9C 95 95 95 95 95 95 95 95 95 95 95 95 145
```

Sum E2 BA D2 AA 9F 9C 5F AA 59 9E 53 AE 03 37 63 45 17A

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
9F00 A5 96 A5 96 A5 9C 95 95 95 95 95 95 95 95 95 150
9F10 A5 96 A5 96 00 96 20 96 20 96 20 96 20 96 20 152
9F20 20 96 20 96 20 A5 20 A5 20 96 20 96 20 96 20 A5 100
9F30 20 A5 20 A5 20 96 20 96 20 96 20 96 20 96 A5 96 173
9F40 A5 96 A5 96 87 20 87 96 A5 95 95 95 95 95 95 10C
9F50 95 9F A5 96 A5 9C 95 95 95 95 95 95 95 95 95 180
9F60 A5 96 A5 96 00 96 20 96 20 96 20 96 20 96 20 1F2
9F70 20 96 20 A5 20 20 A5 20 A5 20 96 20 96 20 96 167
9F80 9C 95 95 95 95 95 95 95 95 95 95 95 95 95 18B
9F90 A5 96 A5 96 87 20 87 96 A5 95 95 95 95 95 95 100
9FA0 95 95 95 95 95 95 96 A5 96 20 96 20 20 20 176
9FB0 20 20 96 00 96 20 20 20 20 20 96 20 96 20 96 186
9FC0 20 A5 20 A5 20 20 A5 20 A5 20 A5 20 96 20 96 18C
9FD0 96 20 96 20 96 20 96 20 96 20 96 20 96 20 96 18E
9FE0 20 96 87 20 87 96 95 95 95 95 95 95 95 95 18F
9FF0 95 95 95 95 95 95 95 96 A5 96 20 96 20 96 A5 96 11B
```

Sum EA 22 F5 95 D9 44 39 29 55 35 45 7B C9 09 43 B0 1FA

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A000 A5 96 A5 96 00 96 20 20 20 96 20 96 20 96 20 106
A010 20 A5 20 A5 20 20 A5 20 20 A5 20 A5 20 A5 20 194
A020 96 20 96 20 96 20 96 20 96 20 96 20 96 20 96 18B
A030 A5 96 A5 96 87 20 87 96 A5 9C 95 95 95 95 95 1F9
A040 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 190
A050 A5 96 A5 96 00 96 20 96 20 96 20 96 20 96 20 1F2
A060 20 96 20 96 20 20 20 20 20 96 20 96 20 20 176
A070 20 20 96 20 96 20 96 20 96 20 96 20 96 20 96 19F
A080 A5 96 A5 96 87 20 87 96 A5 96 95 95 95 95 95 1FB
A090 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 17C
A0A0 A5 96 A5 96 00 96 20 96 20 96 20 96 20 96 20 17C
A0B0 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 1A3
A0C0 20 A5 20 A5 20 96 20 96 20 96 20 96 20 96 20 96 133
A0D0 A5 96 A5 95 95 95 95 95 95 95 95 95 95 95 95 17A
A0E0 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 10C
A0F0 A5 96 A5 96 00 96 20 96 20 96 20 A5 20 A5 20 1C7
```

Sum 7B 9E 7B 1C 22 93 2D 23 72 9C CB 1C B3 22 B2 B0 1AB

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A100 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 11E
A110 20 A5 20 A5 20 A5 20 96 20 96 20 96 20 96 A5 96 142
A120 A5 9E 95 95 95 95 95 95 95 95 95 95 95 95 95 10A
A130 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 1F8
A140 A5 96 A5 96 00 96 20 96 20 A5 20 A5 20 A5 20 A5 106
A150 20 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 11E
A160 20 A5 20 A5 20 A5 20 A5 20 96 20 96 20 96 20 96 159
A170 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 1F1
A180 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 1F1
A190 95 9F A5 96 00 96 20 A5 20 A5 20 A5 20 A5 20 A5 10E
A1A0 20 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 A5 20 11E
A1B0 20 A5 20 A5 20 A5 20 A5 20 A5 20 96 00 9E 95 95 157
```

```
A1C0 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
A1D0 95 95 95 95 95 95 95 95 95 95 95 95 95 95 95 150
A1E0 95 95 95 9F FF 00 00 00 00 2B 43 33 2B 47 52 47 182
A1F0 2B 47 52 32 2B 47 51 52 2B 47 52 2B 47 52 2B 47 180
```

Sum 4B E3 6F 59 6B CF C5 F0 D5 E5 11 E7 9B FE 6F 60 1F6

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A200 41 00 00 00 2B 47 50 2B 41 2B 42 00 00 00 2D 41 144
A210 36 2D 42 52 2D 43 25 43 2D 43 2D 44 45 2D 46 52 182
A220 32 00 00 00 2D 42 51 2D 47 2D 47 2D 47 2D 23 44 159
A230 2D 45 2D 44 52 32 00 00 00 47 36 45 35 46 32 47 138
A240 35 47 52 31 44 32 45 46 47 46 54 45 46 47 32 1EC
A250 2D 41 2D 47 2D 46 2D 45 2D 44 52 32 45 46 47 41 1CF
A260 47 33 52 30 47 47 52 30 2B 43 36 47 46 54 45 44 1E7
A270 35 47 52 31 52 30 43 36 00 00 00 47 32 44 45 44 31 126
A280 45 47 2R 43 52 34 00 00 00 2R 45 32 2B 44 2B 43 10C
A290 42 41 42 31 2B 43 52 34 00 00 00 00 00 FF FF 1F5
A2A0 00 00 00 00 00 18 1B 1B 18 1B 18 1B 18 00 42 102
A2B0 FF 42 5A 52 42 FF 42 00 00 00 00 00 1B 18 00 00 1AB
A2C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 170
A2D0 05 0F 0E 1C 1B 00 00 1B 3B 70 F0 C0 00 00 00 00 1C4
A2E0 0A 1B 1C 9E 0F 03 00 00 00 00 00 1B 3B 70 00 00 104
A2F0 70 3B 1B 00 00 1B 3B 70 FF FF 00 00 00 00 00 1B 196
```

Sum A7 AD 7C 8B A5 96 3C 27 20 99 4A D5 67 5B 99 B2 1FB

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A300 5A 7E 5A 1B BD E7 BD 00 00 BD E7 BD 1B 5A 7E 5A 156
A310 1B 00 00 E0 4E 4B BF BF 4E E0 00 00 07 72 72 154
A320 FD F2 72 07 00 10 3E 06 4E 4E 4E 4E 4E 4E 4E 10B
A330 00 BD FF A5 24 A5 8E 8E 8E 8E 8E 8E 8E 8E 8E 18B
A340 FF 42 E7 00 00 00 00 00 77 77 00 00 00 00 00 1B 12E
A350 1B 1B 00 1B 1B 00 00 00 53 43 45 4E 45 00 00 1E6
A360 44 4F 54 00 00 44 49 41 00 00 42 4F 4E 55 53 00 10C
A370 00 43 43 4F 52 45 00 4B 49 48 48 48 48 48 48 48 10C
A380 47 54 00 20 55 50 20 20 20 20 20 20 20 55 53 48 190
A390 20 42 52 20 4B 45 59 20 00 00 00 20 47 41 4D 45 20 138
A3A0 4F 56 45 52 20 00 00 00 00 00 00 00 00 00 80 00 1BC
A3B0 00 80 00 00 40 02 20 45 00 40 29 09 00 82 04 80 10C
A3C0 44 02 05 01 40 44 04 08 08 08 08 08 08 08 08 08 10C
A3D0 04 8B A1 08 40 12 10 00 B1 01 10 02 40 2B 04 117
A3E0 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 104
A3F0 00 00 00 9C 07 E0 3E 1C 70 86 39 70 70 33 33 38 7B 102
```

Sum EB 24 EC 56 57 5E C5 B4 29 D1 5D BC 2B A2 B2 40 1A7

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A400 76 90 1B 13 CA 00 C2 EC 06 99 4E 50 87 5F B7 1C9
A410 46 1A 92 E4 0B 59 B0 E1 0C 74 8C 66 E6 25 30 CE 144
A420 F7 3D 9A E3 1C 2C 89 6B 7C 32 3B F7 1B 00 CE 176
A430 1B 00 7C 04 00 3B 00 00 00 00 1E 20 1E 20 1E 1B 19A
A440 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 194
A450 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 194
A460 20 1E 20 20 20 20 1E 20 1E 20 1E 20 1E 20 1E 20 1F6
A470 20 20 20 20 20 20 1E 20 1E 20 1E 20 1E 20 1E 20 1F2
A480 1E 1E 20 1E 1E 1E 20 1E 20 1E 20 1E 20 1E 20 1F0
A490 1E 20 1E 1E 1E 1E 1E 20 1E 20 1E 20 1E 1E 1E 1E 1EA
A4A0 1E 20 1E 20 1E 20 1E 20 20 1E 1E 1E 1E 1E 1E 1E 1F0
A4B0 20 1E 20 1E 20 20 1E 20 1E 20 1E 20 1E 20 1E 20 1FB
A4C0 1E 1E 1E 20 1E 20 1E 1E 1E 1E 1E 20 1E 20 1E 1E 1EC
A4D0 20 1E 1E 20 20 20 20 20 20 20 20 1E 1E 1E 1E 20 1F2
A4E0 1E 20 1E 00 00 00 50 41 52 54 20 32 00 00 40 40 152
A4F0 59 2D 41 52 00 00 45 45 45 45 45 45 45 45 45 45 1D0
```

Sum 3B 6A 59 73 74 AF B5 A6 55 1B F1 B9 43 F6 22 46 1A4

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 18um
A500 00 00 97 9B 95 00 9A 3B 9A 00 00 97 9B 95 9B 192
A510 9B 96 97 9B 95 00 9A 3A 9A 00 99 9A 96 9A 96 1A3
A520 00 00 4B 4F 4B 52 55 45 45 44 20 00 20 20 20 20 131
A530 20 9B 96 97 9B 95 95 95 95 95 95 95 95 95 95 95 1E1
A540 9B 9B 9B 9B 9B 9B 9B 9B 95 20 20 20 20 20 9A 1C7
A550 32 9A 00 00 9A 20 20 20 53 50 41 43 45 45 20 20 192
A560 9A 20 20 20 20 20 20 9B 96 00 00 9B 9B 9B 9B 9B 18C
A570 9B 9B 9B 9B 9B 9B 9B 9B 95 00 00 95 95 95 95 95 153
A580 C3 D0 A6 20 5A 84 D9 C4 20 4B 59 2D 43 41 52 5A 154
A590 20 51 20 0E B2 0C 8C B4 CF BD A1 00 00 47 41 116
A5A0 4D 45 20 53 54 41 52 54 20 2D 2D 20 20 20 55 53 1FF
A5B0 4B 20 42 52 20 4B 45 59 00 00 00 ED 5B 36 1B 10C
```

Sum 35 91 0A 76 6A 51 56 5F CB 31 FD 89 43 C4 AB 63 1CA

グラフィック花札

こいこい

MZ-303

■高嶋 晃

グラフィック麻雀：に続く第2弾、「グラフィック花札・こいこい」ゲームが完成しました。

花札は、それぞれの季節をテーマにした花と風物が巧みに図案化して描かれ、日本独特の情緒を持っています。その情緒、風情をできる限り壊さないようにバックしたのがこのゲームです。

花札ゲームの種類はいろいろあるのですが、馬鹿っ花(花合わせ)よりも勝負の「勘」を必要とする、「こいこい」を選びました。秋の夜長を、時間も忘れて夢中になること請け合いです。

さあ、あなたも、華麗な画面上でコンピュータと勝負をしてみませんか？

一. 長 所

●グラフィックをフルに生かしてあります。

花札一枚一枚を忠実に再現するために、縦48×横32ドット一枚の図柄を構成し、48枚とも違う図柄を採用しました(図1)。また、花札の持つ日本的風情を壊さないよう、画面上にはなるべく英数字を表示しないようにしました。

●コンピュータとあなたが「サシ」で勝負する形式をとりました。

二. 準 備

(1) 花札の図柄データをマシン語領域に書き込むため、LIMIT51000 **[CR]** を実行します。

(2) MON モードで***M [CR]** を実行し、SD2F1 番地からデータを書き込んでいきます。データは\$FFFF 番地までの約10K バイト強あります。

(3) データの書き込みが終わると、***S [CR]** にして、

S. ADRS \$=D2F1 E. ADRS \$=FFFF

J. ADRS \$=1300

で、図柄データをSAVEしておきます。

(4) ***J [CR]** J. ADRS \$=1300

でBASICに戻り、BASICプログラムを打ち込んでいきます。

図1 図柄と札コード



※ 最後まで持ち込めば、SAVEしてできあがりです。

三. こいこいの説明

こいこいは2人でやるゲームで、スピード感、スリル、面白さのどれをとっても、馬鹿っ花をしのぐものがあります。

札は48枚すべてを使用し、まず、親・子・場に各々8枚ずつ配ります。残りは山札として、場にふせて積重ねておきます。その後は馬鹿っ花(花合わせ)と同様にして、親から手札を1枚場に出し、場札と絵が合えば取れます。次に山札を1枚めくり、場札と合わせます。その手順を親子順番に行なっていくわけです。

馬鹿っ花と違う点は、得点の対象になるのは役のみで、札自体の点数は勝敗には関係がないという点です。つまり、早く役を作った方が勝ちで、役ができた時点で「勝負」または「こいこい」を宣言します。

「勝負」のときはその時点で精算し、役の得点分がもたら

ます。相手は0文（こいこいの点数単位は文）です。「勝負」を宣言すれば、間違いなく勝てるわけですが、応々にして、ひとつの役ができたときは、次々に役ができてしまうものです。だから、見込みのあるときは少々の文数で止めたほうがいいです。

役ができた時点で、これから役ができそうだと思うたら、「こいこい」を宣言し、ゲームを続行します。ところが、「こいこい」を宣言した場合、もし自分に役ができる前に相手に役ができて「勝負」を宣言されれば、自分の負けとなります。ここが、こいこいの面白い要素で、適切な判断を要求されます。

最後まで、両方に役ができない場合は親の勝ちで、いから親に6文渡します。

四. 花札ゲーム・こいこいの遊び方

RUNすると、最初に、こいこいの簡単な説明が表示されます（写真1）。

Push Any Keyに従って、キーを押すと、マシン語領域のデータを文字変数に読み込んでいきます。

読み込みが終了すると、画面は一転して役札を表示します（写真2、3）。

その後、スペースキーを押すと、さあ、ゲーム開始です。最下行があなたの手札、右2列が場札です（写真4）。親がコンピュータなら、コンピュータから札を合わせ、取札は最上行に並べられます。親があなたなら「ドレ ヲ キリマスカ?」の質問に数字で入力してやってください。もし、あなたが切った札と同月の札が場に複数枚でなければ（たとえばあなたが桜の1点札を切り、場に桜の5点札と20点札があった場合）、「何点札と合わせますか?」ときいてくれるので、5点札なら5、20点札なら20と入力してください。

あとはどんどん役を作っていけばいいのです。コンピュータに役ができれば、「こいこい」か「勝負」かを自動的に判断して宣言してきます。あなたに役ができれば、「勝負=0、こいこい=1?」と聞いてくれるので0か1を入力してください（写真5）。

サービス機能として、コンピュータまたはあなたが「勝負」を宣言すれば「野球拳」のメロディが、「こいこい」を宣言すれば「しよじょ寺のためき林」のメロディが美しく(?)流れてきます。なぜ、こいこいがしよじょ寺なのか……は、歌詞を口ずさんでもらえばわかります。ね??

もうひとつのサービス機能として、ゲーム開始後、好きなところで[C]のキーを押せば、その時点のハード・コピーがとれます。もちろん、ハード・コピーが終了次第、ゲームは続行されます。たとえば、五光などができたときは記念に残しておきましょう。

ゲームは12回で終了となります。くれぐれも、コンピュータのような単純知能の機械には負けないでください。もし、あなたが負ければ、コンピュータが笑うように作っています（これは冗談）。

役は、表示例でもわかるように、五光、四光、三光、赤タン、青タン、猪鹿蝶のみ登録してあります。このほかに、月見酒、花見酒、タネ、タン、カスなどがあるのですが、地方によっては役にしないところもあり、また月見酒などはできやすい

写真1

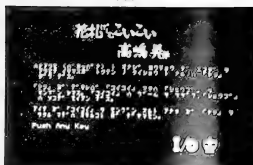


写真2



写真3



写真4



写真5



のであてて採用しませんでした。猪鹿蜂も役に入れない地域もあるようですが、これは小生の好みで採用しました。

また、役の得点数もいろいろあるようですが、一応、表1の規準でいきました。

五. プログラムの説明

概略のフローチャートを図2に示します。

絵札その他のグラフィック・データは16進数で\$D2F1から\$FFFFまでの約10Kバイト強のメモリ・エリアに納められており、プログラムの320行から540行まで、16進データを文字変数に代入しています。なにせ、10Kバイト以上ありますので、代入には少々時間がかかります。

48枚の絵札は各々、縦48×横32ドットで構成され、その上半分(縦24×横32ドット)をF1\$(1)~F1\$(48)に、下半分をF2\$(1)~F2\$(48)に代入されます。各々の札には番号(札コード)が割り当ててあるので(図1)、たとえば、桜の20点札(札コード=45)を表示するにはF1\$(45)とF2\$(45)をPATTERN文で実行すればいいのです。1枚の札を2つの文字変数に分けたのは、PATTERN文の制約によるもので、取説には書いてありませんが、PATTERN文の積み重ね段数は24ドットがMAXなのです。ということは、どうもシャープさんは漢字表示のためにPATTERN文を設けたようです(漢字は通常16×16ドット、または24×24ドットで表現するから)。

820行で親を決めています。変数OYA=0なら親はコンピュータ、1ならあなたです。

1020行から1300行まで、札を配って(札コードを表2の各変数に割り当て)ています。

1520行から1730行まではコンピュータの札合わせ、1800行から2015行まではあなたの札合わせを行なっています。両者の相違点は、あなたは指定した切り札と場札が含まれているかどうかの判定だけを行なっているのに対し、コンピュータは手札と場札のすべての組み合わせの中から最適なものを抽出する必要があるという点です。

グラフィックのデータがこのプログラムの「顔」とするならば、この「最適組み合わせ抽出」は「脳ミソ」にあたる重要な部分と言えます。

改良して、コンピュータをより強くしてみたいと思われる人もおられるでしょうから、この部分のアルゴリズムを記述しておきます。

まず、必要なテーブルが2つあります。ひとつは合致したすべての組み合わせのデータを一時、貯える配列変数CO(i, j)と、あとひとつは役のデータを貯えるTA(i, j)です。変数の内容の意味は表3をみてください。

「最適抽出」の際の優先順位は図3のフローチャートを見てもうば理解できると思います。5500行から5680行でCO(i, j)テーブルを作成し、何対の役の組み合わせができたかをCCに入れてリターンします。その結果を5700行から5910行で判断し、最もいい組み合わせを選択してTA(i, j)テーブルに反映しています。

ほま、人間が札合わせをするときの思考を再現していますが、大局的な見地になったる思考はできません。たとえば、手札の中に赤タンノ元になるカス札が3枚そろっているから、他のカス札から順に切っていく……というようなことは考慮できません。これは今後の課題です。幸いにして、まだメモリ・エリアは約7Kバイト余っているのです。どなたか挑戦してはどうですか？

図2 概略フローチャート

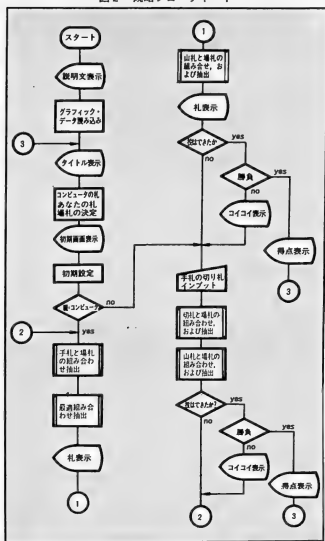


表1

五 光	四 光	三 光	青 丹	赤 丹	猪 鹿 蜂
15 文	10 文	8 文	6 文	6 文	5 文

表2

F(i, j)	1	2	3	4
コンピュータ手札	あなた手札	場札	コンピュータ場札	あなた取札
1~48の札コード				

表3

CO(CC, YC)		-CC				TA(XA, Y) -XA						
		0	1	2	3	0	1	2	3	4	5	6
Y	C	役なし	役あり									
		役なし	役あり									
		役なし	役あり									
		役なし	役あり									
Y	C	F(0, j)のj0の値										
		F(2, j2)のj2の値										
		TA(XA, Y)のXAの値										
		TA(XB, Y)のXBの値										
注). プログラム実行前にイニシャル・セット												

		五	四	三	青	赤	黄	緑
		光	光	光	丹	丹	丹	丹
0	可能性フラグ	9	0	0	0	0	0	0
	コンピュタ役までの残残数	9	5	4	3	3	3	3
1	あなた役までの残残数	9	5	4	3	3	3	3
	可能性フラグ	1	2	3	4	5	6	7

1-コンピュタ役が持っているのみ
 2-あなたが持っているのみ
 3-持合
 4-コンピュタ側の役決定
 5-あなたの役決定


```

D6B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D6C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D6D0 00 00 00 00 00 00 00 00 00 00 00 02 06 7F 0F 1CC
D6E0 06 02 00 01 07 1F 7F 7B 72 06 06 0C 0C 0C 0C 1DD
Sum 06 02 00 01 07 1F 7F 7B 72 06 06 0B 12 82 4B 1B 1A9

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
D6F0 00 00 01 01 03 03 0F BF 03 01 B1 8B 3B 1B 0C 0E 14C
D700 0C 1B 1B 1B 1B 1B 1F 00 00 00 00 00 00 00 00 00 1FC
D710 80 80 07 33 60 40 0F 00 00 00 00 00 00 00 1B FC 1FD
D720 00 00 00 00 00 00 0C 0C 1C 1C 3F FF FF 1C 0C 0A 1CD
D730 0C 1E 7F EC BC 0C 0C 0C 1C 3F FF FF 00 0E 07 07 06 12D
D740 04 86 C6 04 0C 0B 04 0A 0C 0C 0C 0C 0C 0F 07 14E
D750 00 00 00 00 00 13 F1 1D 0B 08 00 00 00 70 38 04 1ED
D760 1B 1C 0C 0C 1C 1C FC FB 00 00 04 84 80 80 80 1DA
D770 80 80 80 08 08 08 08 08 03 00 00 01 01 01 09 00 1B4
D780 00 00 00 00 00 00 00 00 00 00 00 00 00 00 40 140
D790 04 42 C0 40 C2 42 3E 00 60 7F 1F 03 03 03 06 1B2
D7A0 00 00 00 00 16 20 60 F0 FF 7F 9F 00 00 00 00 100
D7B0 00 00 00 80 80 80 80 80 80 04 07 07 03 03 03 1E2
D7C0 33 F1 E1 0C 00 00 00 00 00 00 00 00 00 00 00 1C5
D7D0 00 00 00 80 80 80 80 80 80 80 C2 C6 8C 7C 3B 18 1CA
D7E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7B 3C 152
Sum 57 00 1B 0D 11 24 CA FB AE D7 7B 72 72 02 32 12 171

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
D7F0 1E 1E 1E 0C 00 00 00 00 00 00 00 00 00 00 00 17F
D800 FF FF FF 9F 81 C0 E1 FF FF FF FF FF FF B3 C1 E2 E6 14F
D810 EF CF CF C3 93 99 FF FF FF FF FF FC EC 0C 1D 1C 1FD
D820 9C 9E BE 9E 9C 61 F3 FF FF FF FF 8B 43 7F 9F DF 1E5
D830 FD 7D 58 55 CD 0C 0F 00 00 00 00 00 00 00 00 00 1FC
D840 C3 F3 F3 73 73 73 EF FF FF FF FF FF 9F 9F 9F 16F
D850 81 39 2F 21 21 21 FF FF FF FF FF FB FB FB FB 1FD
D860 FC FB F3 F1 FB FC FF FF FF FF 7F 33 03 03 03 03 1C2
D870 1B 84 BC 9C 3C 7F 93 FF FF FF 7D C0 63 03 F5 1B4
D880 F3 E7 E7 1B4 00 00 00 00 00 00 00 00 00 00 00 18A
D890 EF EF 07 63 E5 47 0F FF FF FF FF 67 00 01 E7 E6 14A
D8A0 E7 C7 CF CC DC FE FF FF FF FF FF FF FF FF 1F 10B
D8B0 9F FF FF FF F7 07 0F FF FF FF FF FF FF FB FC FB 14E
D8C0 FF FF FF FF FB FC FF FF FF FF FF FF FF 7B 19 157
D8D0 7F 9F 9F FC 1E 10 0F FF FF FF FF FF FF FF 7F 9F 157
D8E0 FB 9F 9F D9 3F 3F 3F FF FF FF FF FF FC C1 0E 1D7
Sum 09 2C 3C 76 39 F9 EA OE F1 F0 E0 3E EB BF 87 C6 1D4

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
D8F0 FF FF FF D9 9F 80 C0 E1 FF FF FF FF FF FF FF FF 15F
D900 CF CF CF E6 E1 F1 7F FF FF FF FF FF FF FF FF FF 15A
D910 DF CF CF CF 0F 0F 0F FF FF FF FF FF FF FF FF FF 154
D920 FD FD FD FD FD FD FD FF FF FF FF 24 84 EC 80 3C 15C
D930 8B 80 A3 35 AB 8D 39 3B FF FF FF FF 3F 3F 3F 3F 151
D940 FF 0F 3F 2F 2F 3F 3F FF FF FF FF 9F 9F 9F 9F 1A7
D950 F3 F3 F3 F3 F3 F3 FF FF FF FF 00 42 01 F3 F3 02 155
D960 F3 F3 F3 F3 F3 F3 FF FF FF FF 00 42 01 F3 F3 02 155
D970 FF FF FF FF FF FF FF FF FF 00 42 01 F3 F3 02 155
D980 00 10 7C 10 1C 02 02 04 00 0B 09 08 08 09 05 1F7
D990 00 00 00 40 00 80 00 00 10 7C 10 2B 20 18 1B 1CC
D9A0 00 00 20 40 40 40 40 00 7E 60 60 60 60 7C 1A5
D9B0 00 00 00 00 00 80 00 00 00 00 00 00 00 00 00 1A5
D9C0 00 10 AB 10 00 00 00 C0 3C 0B 0C 3C 02 02 0C 12B
D9D0 00 04 0B 10 20 10 08 04 00 20 7B 26 44 4E 15 0C 169
D9E0 00 04 00 40 4C 18 24 18 04 04 5F 44 44 4E 55 2C 1DC
Sum 42 92 9A 65 FC C9 EF 96 F7 70 B8 23 EB 3B 40 F2 1B4

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
D9F0 00 10 7E 10 1C 02 04 1B 00 00 3C 66 66 66 66 112
DA00 66 66 66 66 66 66 66 00 00 1B 3B 7B 5B 1B 156
DA10 1B 1B 1B 1B 1B 1B 1B 7E 00 00 3C 66 66 66 66 106
DA20 0C 0B 10 10 30 62 66 7E 00 00 3C 66 66 66 66 04 12A
DA30 0B 04 00 00 00 00 00 00 00 0C 1C 2C 2C 2C 1AE
DA40 6C 6C 6C 7E 0C 0C 0C 00 00 00 00 60 60 60 7C 1A5
DA50 7E 46 06 06 06 66 7E 3C 00 00 3C 66 66 66 66 00 11E
DA60 7C 7E 66 66 66 66 66 3C 00 00 7E 7E 62 62 40 06 154
DA70 06 0C 0C 1B 1B 10 30 00 00 3C 66 66 66 66 3C 1EE
DA80 3C 66 66 66 66 66 66 3C 00 00 3C 66 66 66 66 66 116
DA90 6E 7E 06 66 66 66 66 00 00 00 00 00 00 3F 0B 1AD
DA90 6E 7E 06 66 66 66 66 00 00 00 00 00 00 00 00 1AD
DA90 04 06 03 01 01 03 0C 30 00 00 C0 C2 FF 1B 1AF
DAB0 1B 10 70 60 C0 7B 1F 07 00 00 C0 00 00 00 00 1F6
DAC0 00 7B 40 42 43 43 49 4E 4E 45 41 01 59 79 71 1D7
DAD0 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 1B5
DAE0 7E BC 1C 84 C0 C0 40 00 00 00 00 00 00 00 00 1B5
Sum 84 02 7D F4 CD 5B 7F 84 C6 4A 39 71 83 67 B4 AF 1E9

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DAF0 FA FF 00 99 7C 58 43 AF 7E AC 00 00 00 00 00 00 185
DB00 41 40 4C DE 00 00 00 00 00 00 00 00 00 00 00 1BE
DB10 5F 6D 21 61 41 1D 1F 45 E9 C9 1D 1D 09 11 11 12B
DB20 09 5D 7D 5B 41 41 41 43 43 41 40 40 40 40 40 144
DB30 40 40 40 40 40 40 40 7F 00 60 00 06 FA 60 80 1EA
DB40 B6 0F 04 00 00 00 00 00 00 00 00 00 00 00 00 18A
DB50 00 6C 40 40 4C DE 6C 40 40 4C DE 6C 40 40 40 1D8
DB60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 176
DB70 87 65 19 21 21 41 59 45 25 19 09 01 01 01 01 164
DB80 00 7F 40 40 40 40 40 40 40 70 5C 4E 47 43 43 169

```

```

DB90 47 42 40 40 41 41 41 47 72 FF 00 00 00 00 00 00 17C
DBA0 00 00 00 00 00 19 B1 97 02 66 04 BF D5 BA C5 12 1D6
DBB0 71 FF 00 00 00 00 00 00 00 00 01 01 03 B3 67 23 0B 1D6
DBC0 16 2B 35 EA 59 57 05 A6 4E FF 01 01 1F 70 71 81 1A3
DBD0 E1 C1 C1 81 81 81 81 81 01 01 01 81 83 CF 1D 15C
DBE0 79 5B 3C 4E 4E 47 47 63 73 79 5C 4F 43 79 79 5C 1A2
Sum CC 29 5B 0D 39 6D 32 B6 24 92 B7 32 FB 66 2A CB 11D

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DBF0 4E 47 47 47 47 43 43 7F FF 00 72 72 3A BA BB 1B3
DC00 FC FC FC FC 7E FE FF FF FF 7F 7F 7F 7F 7F 7F 1A2
DC10 00 BC 3C 3F 3F 3F 3F 7F 7F 7F 7F 7F 7F 7F 7F 1A0
DC20 FF FF FF FF FF FF FF FF FF 00 71 E1 C1 81 8F 1D 11F
DC30 39 31 71 F1 E3 E7 8C 80 79 F1 F1 E9 C1 81 BF 166
DC40 00 7F 40 40 40 44 4A 4A 4A 4A 5B 66 59 41 41 40 17B
DC50 41 43 41 40 40 47 4E 4E 4E 00 00 00 00 00 00 160
DC60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 173
DC70 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 152
DC80 74 3C 34 15 FC 87 19 3B 7F 01 01 01 01 01 01 14A
DC90 01 C1 21 C1 01 B1 41 B1 21 61 61 61 61 61 61 142
DCA0 1D 40 40 40 40 46 45 46 46 46 46 46 46 46 46 105
DCB0 40 40 40 40 40 40 40 40 7F 00 66 00 02 35 35 32 116
DCC0 1E 0F C8 4A 4A 4A 44 00 00 01 03 03 00 00 00 1A8
DCE0 00 60 40 71 70 31 FE 04 1A 71 F0 E0 23 3B 39 166
DCE0 3B FF C1 0E 0C 00 00 00 00 00 00 00 00 00 00 1D0
Sum E2 54 2E 5B 54 9D BC 5A EB 10 0B F3 00 17 2A AB 17B

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DCFO 79 39 51 BF 47 2F 91 A9 75 63 E1 41 39 07 01 FF 17C
DD00 00 7F 40 41 4F 5F 4F 47 47 47 40 40 40 40 40 152
DD10 40 40 43 43 41 59 5B 5C 7E 6F 00 80 C0 F0 FF FB 1ED
DD20 F0 F0 F0 80 00 40 40 7F 00 00 00 00 00 00 00 152
DD30 10 FF 00 00 00 7F 81 82 5C 5C 5C 5C 5C 5C 5C 152
DD40 9B 1C 0F 07 0F 0F 2F 3F FF 01 01 01 01 01 01 1A4
DD50 5F 9B 91 51 D1 61 01 01 01 01 01 01 01 01 01 18A
DD60 5F 7F 7E 7C 7C 7D 7F 41 41 41 47 5F 7F 7F 7F 155
DD70 7F 40 40 40 40 40 40 7F 00 00 00 00 00 00 00 157 12E
DD80 1A 1F 12 7E 22 FD C1 81 81 01 01 01 01 00 00 160
DD90 00 20 60 1D 4B 36 F2 52 0F FF 7F 8C 04 04 07 FF 12E
DDA0 FF FF FF FF FF FF 00 FF 00 FF 00 7D 19 31 21 21 1F0
DDB0 7F 3F 3F 3F 3F FF FF FF FF C1 C1 C1 C1 C1 1F 176
DDC0 00 7E 41 41 41 43 43 47 47 47 40 40 70 5E 41 171
DDD0 64 55 71 43 44 40 46 46 7B FF FF FF FF FF FF 166
DDE0 1C 1B 2B 2B 48 44 84 24 4C 46 F3 B9 00 5F 60 70 1F5
Sum 6B E7 0F 1A 33 99 2C 76 94 A9 C1 27 EA 3B 51 91 1F3

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DDF0 F0 FF 00 80 C0 E0 F0 70 30 0B 00 00 01 01 00 11A
DE00 00 00 81 87 9C 0F 00 00 00 FF 01 01 01 01 01 1B8
DE10 01 01 01 01 F1 00 B3 C9 43 43 41 51 51 51 51 1E8
DE20 23 40 41 47 5F 63 43 41 41 59 54 53 4B 4B 45 174
DE30 44 7D 5B 44 46 42 72 0F 00 FF FF FF FE FE FE FF 15C
DE40 FF CF 03 00 00 C0 6F 30 8B 52 41 9B 37 60 5F 1DC
DE50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1C0
DE60 80 80 80 80 80 00 00 00 00 00 67 5D F1 81 01 01 179
DE70 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 1FF 0E
DE80 00 7F 40 40 40 40 40 40 40 40 40 40 41 43 43 47 110
DE90 7F 40 41 43 47 7E 7D 7A 7A 5A FF 00 00 5A 25 134
DEA0 2B 07 07 83 FB FE FE 0F 4F 7F FE D5 AA 5A 5A 157
DEB0 AA FF 00 00 40 54 8A 5A 5A C0 C0 80 7E FF FF 18D
DEC0 FE 80 00 40 A8 54 AA 5A AA FF 01 01 01 01 01 16B
DED0 01 01 01 01 01 01 01 01 01 01 01 01 01 71 7F FF 174
DEE0 FF 45 4A 65 7D 7A 7D 7A 7D 7C 79 7C 49 47 43 1E5
Sum EB 09 D3 AE DA BC EC 9C 34 4F 25 32 D5 49 4C 00 1D7

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DEFO 47 4E 46 47 4F 4C 5B 7F 00 55 AB 5A 5A 5A 5A 104
DF00 AA 55 AA D5 FF FF FF 7F 7F 7F 7F 7F 7F 7F 7F 154
DF10 00 55 AA 5A 5A 5A 5A 5A 5A 5A 5A 5A 5A 5A 5A 18A
DF20 FB C0 C0 80 00 00 00 FF 00 7F 8F 7F 8F 7F 7D 1140
DF30 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 18A
DF40 00 7F 40 40 50 5B 5B 7B 5B 7B 5B 5B 5B 5B 40 147
DF50 4E 7C 4E 4E 7E 4F 4B 4B 70 FF 00 00 00 00 00 132
DF60 00 00 00 00 00 00 00 00 00 00 00 00 00 80 80 100
DF70 7F 00 00 00 00 00 00 00 00 00 00 00 00 0C 0C 1BF
DF80 1C 24 1C 0C 0C 0C 0C 0C 0C 0C 01 01 01 01 01 1C1
DF90 81 C1 C1 C1 D9 E5 E5 D9 C1 C1 C1 D9 E5 E5 D9 C1 180
DFA0 C1 42 47 40 72 52 40 45 44 4B 4B 74 42 4F 45 60 16E
DFB0 40 45 65 59 41 40 40 7F 00 00 00 00 8B 74 44 33 00 157
DFC0 80 40 71 09 B5 05 09 31 21 C1 FF FF 1F 0F 03 FF 176
DFD0 01 01 01 01 BF BF 0E 1C 22 C1 C1 C1 C1 C1 C1 18A
DFE0 FF C7 C7 C7 FC FC FF FF FF 00 C1 C1 C1 81 01 01 156
Sum BD 30 94 7D F3 DE 32 7E 76 13 5C 13 6A 02 2C 32 101

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
DFF0 39 39 E7 21 21 2D F3 B0 01 01 01 01 01 01 01 1FF 10E
E000 00 7F 40 40 40 40 40 40 40 40 40 40 40 40 40 1FF
E010 40 40 40 40 40 40 48 54 FF 00 00 00 00 00 00 00 16F
E020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 16F
E030 00 FF 00 00 00 00 00 01 02 02 07 07 17 0F 0F 176
E040 1F 1F 5F 3F 3F 3F 3F 3F 3F FF 01 01 01 01 01 176
E050 B1 81 B1 C1 D1 E1 E1 E1 F1 F1 F1 F1 F1 F1 F1 F1 1D0
E060 FD 5C 7C 7D 7E 7E 7F 7F 7F 7F 7F 7F 7F 7F 7F 144

```

データ・ダンプ・リスト

E070	7F	7F	7F	7F	7F	7F	7F	00	00	00	01	00	B0	00	00	179
E080	40	90	AB	FB	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	110
E090	00	7F	7F	7F	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	171
E0A0	FF	FF	FF	FF	FF	FF	FF	00	FD	FD	FD	FF	FF	FF	FF	1EB
E0B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1FO
E0C0	00	7F	40	40	40	40	40	40	40	40	40	40	40	40	40	166
E0D0	40	40	4E	4E	42	42	42	42	FF	00	00	00	00	00	00	16E
E0E0	09	70	46	4C	23	12	22	15	08	01	07	36	4B	B7	31	1BD
Sum	1C	AE	39	9C	4F	5A	3A	61	99	E7	03	43	CD	4C	5C	159

[illegible]

turn	RE	CD	95	BA	4C	F1	17	FA	2D	0B	90	31	45	BA	7B	94	1B0
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
E160	7F	7F	7F	7F	7F	7F	7F	7F	7F	00	00	00	00	3F	FF	FF	13um
E170	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	14um
0010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	15um
E220	FF	FF	FF	FF	FF	FF	FF	FF	FF	01	01	01	01	01	01	01	16um
E230	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	17um
E240	00	7F	40	40	40	40	40	40	40	40	40	40	40	41	43	41	5B 1C
E250	5C	4C	4C	47	43	41	43	46	5F	FF	00	03	03	01	08	1C	1E
E260	0E	01	00	0E	0E	0E	0E	0E	0E	0E	0E	0E	0E	0E	0E	0E	1F
E270	FB	90	00	1B	9C	9B	80	40	46	5C	0E	46	4C	D0	40	7E	1F3
E280	FC	7B	7C	7E	5F	5F	5F	5F	01	01	01	01	01	01	01	01	1C
E290	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	160
E300	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	170
E320	4F	47	48	44	44	44	45	7E	00	FC	FF	FF	FF	FF	FF	FF	16F
E2C0	FF	7F	1F	0F	03	00	07	0B	96	53	D6	6B	87	0E	7F	16A	170
72D0	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	127
72E0	40	40	40	40	40	40	40	00	00	00	F9	FD	FD	FF	FF	FF	128

[illegible][illegible]

Sum	B9	11	DB	08	46	B4	05	4A	E4	04	9F	FD	E1	5B	14	10	17A
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
E4F0	5C	5C	5E	4F	4F	47	7F	00	AA	5D	E2	09	F8	FC	FC	1A3	
E500	FC	FC	FC	FE	FE	FE	FE	FF	FF	7F	7F	FF	FF	FF	FF	FF	1A3
E510	00	AA	57	4A	59	43	47	0F	0F	BF	0F	5F	5F	1F	3F	12F	
E520	7F	7F	7F	BF	BF	BF	BF	0D	0D	99	B9	31	F1	F1	E1	137	
E530	E1	EF	EF	F9	E1	CF	9B	B9	F1	E1	E1	C1	C1	C1	FF	194	

E540	00	7F	40	40	40	40	40	41	40	40	40	41	40	40	101
E550	40	00	5B	54	CA	5A	67	5B	40	00	00	3B	24	1C	CB
E560	29	DE	00	C4	34	CE	03	62	5F	36	0C	40	40	40	156
E570	52	01	00	00	00	00	00	00	00	00	00	00	91	60	156
E580	12	0F	01	55	9B	34	97	CF	0F	01	01	01	01	01	174
E590	01	01	01	01	01	41	99	95	99	81	01	61	91	01	13F
E5A0	01	71	7B	5B	6A	74	7C	5C	43	77	5B	66	77	74	16A
E5B0	4D	5F	59	73	61	40	7F	00	B1	40	26	6F	47	5F	FD
E5C0	99	B3	0E	3F	64	CE	0C	19	FF	91	09	99	DC	84	0F
E5D0	00	CB	38	17	3B	23	20	E8	9E	12	29	28	10	FF	15A

E5E0	08	14	95	C9	C0	00	00	FF	00	B1	01	01	B1	01	01	01	140
Sum	F0	11	33	69	0D	15	4A	67	93	78	95	39	77	C7	65	182	
AdF0	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	122
ESF0	01	01	B1	41	01	41	99	E5	99	B1	B1	41	01	B1	01	FF	122
E5F0	00	7F	40	40	40	40	40	40	40	40	40	42	41	40	40	40	122
E610	41	41	47	47	43	41	01	41	43	FF	00	00	1F	0F	00	00	187
E620	00	7F	FE	01	01	08	DC	FC	FE	FE	FE	FE	FF	FF	30	10	E94
E630	10	FF	00	00	00	0E	9A	93	93	92	92	FE	3F	1F	10	30	186
E640	00	7F	40	40	40	40	40	40	40	40	40	40	40	40	40	40	122
E650	01	01	01	01	C1	E1	01	01	21	61	E1	F1	F1	F1	F1	DC	1DC
E660	FF	42	5F	54	65	65	64	63	56	49	48	48	4C	43	47	47	1A1
E670	4F	5F	4E	4E	44	40	40	7F	00	00	1E	F2	A6	55	EB	1E	1A3
E680	02	8F	F9	41	C1	E2	FE	FB	FB	FB	F0	10	00	00	00	FF	153
E690	00	7F	40	40	40	40	40	40	40	40	40	40	40	40	40	40	122
E6A0	07	06	02	00	00	00	00	00	00	B1	01	01	01	01	01	F9	18D
E6B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	3F	07	01	01	01	01	FF	13E
E6C0	00	7F	40	40	40	40	40	40	40	40	40	41	4F	40	40	40	122

ADD	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0	160
ADD	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0	160
Sum	D4	62	3A	1A	A6	BE	62	C2	4F	BD	6B	B1	EA	8A	41	E6	14C
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+18B
E6F0	00	FF	00	3E	11	0A	39	63	70	FE	FD	FA	F7	F8	F9	F8	1CB
E700	3C	0C	04	00	00	00	00	00	FF	01	01	9F	75	4D	99	147	
E720	4D	27	03	31	00	00	00	00	00	00	00	00	00	00	00	00	1E6
E740	00	00	00	00	45	45	45	45	4C	47	41	4C	5A	70	41	73	4E
E760	40	63	50	70	43	46	46	7C	00	00	00	00	00	FE	04	08	1BD
E780	08	44	62	6F	40	27	3C	01	E3	BE	50	2E	03	C1	30	FF	143
E7A0	00	01	3F	1F	27	43	33	05	0F	02	04	08	1D	DC	00	80	155
E7C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	155
E7E0	09	49	A5	45	45	25	15	09	01	01	01	01	01	01	01	FF	13E
E800	70	7F	40	40	40	40	40	40	40	40	40	40	47	43	41	111	
E900	40	41	53	7B	7F	7F	7F	7F	7F	FF	00	00	00	20	15	0A	109
E920	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	109
E940	FF	FF	00	00	00	00	44	AA	54	AB	4C	00	C0	00	1E	BF	11F
E960	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	12C
E980	7F	FF	9F	9C	C0	6A	55	2A	55	FF	01	01	01	01	01	01	12C

[illegible]

EELO	20	00	00	00	00	00	00	FF	00	01	01	01	01	01	61	146
Sum	BF	1E	74	0B	43	13	56	26	EC	5B	82	0F	E2	4D	91	198
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	15um
E900	01	95	47	CF	7B	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F
E910	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F
E920	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E930	E7	FF	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E940	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E950	01	01	01	01	05	0B	08	0F	5F	3F	BF	7F	7F	7F	FF	106
E960	FF	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F
E970	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F	7F
E980	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
E990	FF	FF	FF	FF	81	85	83	83	FB	FF	FF	FF	FF	FF	FF	FF
E9A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
E9B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
E9C0	7F	7F	40	40	43	44	7B	63	67	55	4B	58	00	87	42	101

[illegible]

データ・ダンブ・リスト

```
F410 00 00 00 00 00 00 F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F420 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F430 FD FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F440 00 7F 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40
F450 46 42 50 4E 41 40 47 43 41 FF 00 00 00 00 00 00 00 00 00 00 17E
F460 01 01 00 00 00 00 00 00 31 70 60 41 80 80 40 F9 F0 E0 14E
F470 F0 FF 00 00 00 00 00 00 31 70 60 41 80 80 40 F9 F0 E0 14E
F480 3E FC FC 7C 7E 7F FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F490 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01
F4A0 0E 43 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F 4F
F4B0 00 7F 42 50 4E 41 40 47 43 41 FF 00 00 00 00 00 00 00 00 00 00
F4C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F4D0 00 7F 7F 7F FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F4E0 00 04 7E 04 0C 14 24 FF 00 F1 F9 FF FF FF FF FF FF FF FF
Sum 10 5E 1D 7B 77 9D 9C D1 72 1B 0C CE FC 5F EC 6D 1A2
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
F4F0 E1 E1 E1 C1 C1 C1 B1 01 01 01 01 01 01 01 01 01 01 01 01 01
F500 00 7F 42 50 4E 41 40 47 43 41 FF 00 00 00 00 00 00 00 00 00
F510 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F 5F
F520 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
F530 50 50 50 50 50 50 50 50 50 50 50 50 50 50 50 50 50 50 50
F540 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08
F550 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9 F9
F560 F7 4F 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44
F570 50 40 60 50 40 40 40 40 7F 00 00 00 00 00 00 00 00 00 00
F580 08 05 05 03 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00
F590 00 4A 82 82 00 80 81 41 41 51 51 A9 98 98 98 98 98 98 98
F600 46 46 02 02 00 00 00 FF 07 F7 F7 F7 F7 F7 F7 F7 F7 F7
F610 4F 4F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F 9F
F620 00 7F 40 40 40 41 41 42 44 40 40 40 40 40 40 40 40 41 122
F630 23 26 2C 1B 04 04 82 0E 72 80 40 00 00 CF 41 42 1A9
Sum EA A3 B0 45 AB 89 06 3D 07 91 D0 26 A7 76 1B A5 16E
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
F6F0 C2 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
F700 11 02 04 1F 00 40 60 7C CB 90 39 20 12 15 CB 08 08 14B
F710 39 91 B1 81 07 8F 47 C1 01 E1 41 83 05 49 F9 69 1CD
F720 08 41 41 40 40 40 51 5F 4F 5F 4F 45 78 4B 18D
F730 51 50 48 49 46 40 47 00 01 80 FF 00 00 00 00 00 00 177
F740 87 80 40 80 00 00 00 00 00 00 80 80 00 00 00 00 00 00
F750 00 00 C1 E0 00 0C 1E 00 00 79 35 01 42 F5 6C 18B
F760 04 02 07 07 03 01 00 FF 0F F3 D1 17 2F 25 14C
F770 0F 9F 00 01 01 C1 E1 01 01 01 01 01 01 01 01 01 01 01
F780 00 7F 41 4A 49 47 40 41 60 70 78 7C 58 53 66 105
F790 40 46 71 46 46 46 46 46 46 46 46 46 46 46 46 46 46
F800 03 41 C9 CC 4C 8C 0C 8B 5A 55 5E 5E 5E 5E 5E 5E 5E
F810 FC FF 02 74 5F 2B 2F 00 81 B1 83 87 3B 35 5A 18B
F820 AC 5A 5A 5C 0C 8C 7C 78 7F 8F 89 25 85 01 79 164
F830 C1 B1 B1 01 01 01 01 01 01 01 01 01 01 01 01 01 01
F840 E1 4E 4F 7F 4F 3F 4C 47 60 7F 4C 4F 4B 4B 4B 4B
Sum 35 AD F3 66 DE 5C 00 8A 45 E4 3B 2C 85 22 F1 66 1BA
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
F8F0 58 70 41 47 5C 71 43 7F 00 7E 7E 7E FD FD 18C
F900 CD 79 8E 88 8E 8F 1C 71 C7 1C 70 00 00 FF 12B
F910 00 79 F9 F9 1B CB 23 EF 08 83 0E 18 70 C3 7E 1E5
F920 42 C2 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42
F930 CD CD F5 19 71 C3 02 07 07 0F 1F 3F 3F 7F FF 136
F940 00 7F 40 40 40 41 4E 42 4C 70 00 40 40 40 40 40 40
F950 40 40 40 40 40 44 4A 4B 46 FF 00 20 6E 91 80 21F
F960 52 52 31 09 0E 39 20 50 51 23 03 03 03 00 00 E0 1E3
F970 3E FF 00 61 30 06 00 00 8B CB 1C 0B 08 3B 4F 16C
F980 80 9C 98 88 33 01 01 00 00 FF 01 B1 01 01 05 08 13A
F990 08 05 35 40 35 05 05 85 F5 1D 07 01 B1 C7 CF 1D6
F9A0 31 42 42 43 47 4E 4B 40 43 42 46 44 45 45 5B 67 17E
F9B0 59 41 42 41 40 40 40 7F 00 43 70 81 80 81 C0 184
F9C0 70 1F 01 84 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A 8A
F9D0 00 07 79 0E 08 01 00 10 01 00 81 C0 81 89 8A 1C5
F9E0 B3 FE 0C 98 16 0F 00 FF F9 CD 87 3B 8D 49 E1 1E8
Sum 0C FC BF 4B 86 44 FE B0 11 B5 ED 1E 65 27 41 CA 1C2
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
F9F0 31 99 00 81 41 B1 8D 90 10 9D D1 E1 39 0F 07 FF 1E
FA00 00 7F 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40
FA10 40 40 40 41 43 41 40 40 40 40 40 40 40 40 40 40 40
FA20 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F
FA30 FF FF 00 00 01 03 07 07 8F BF 17 C8 42 7F 5C 1F8
FA40 BF CD 86 DA 7B C2 3E 08 91 FF 01 00 3D 7B FF 1D
FA50 FF FF FF FD 7D 19 00 00 CB 79 71 49 C9 49 20 18E
FA60 FF 4D 47 43 41 40 40 4C 4B 48 4C 48 49 49 50 152
FA70 56 49 72 5E 40 43 40 7F 00 FF FF 1C 08 00 00 1D2
FA80 40 BF 42 4A 4C CB 28 00 80 E0 8E B1 BF 20 FF 1E
FA90 00 FF 9F 3E 7E 0C 00 02 07 06 19 30 4A 72 182
FAB0 0C 02 01 01 00 00 31 60 FF FF FF 31 0B 71 171
FAC0 1B 67 8B 87 47 A3 7D 51 61 01 01 01 01 01 01 01
FAD0 00 7F 40 40 40 40 41 42 41 40 40 5A 4A 4D 13E
FAE0 08 4B 4B 4B 4B 50 51 51 5F 00 00 49 15 2C 183
FAF0 00 89 49 55 60 01 01 07 0C 3B 08 78 40 80 00 18D
Sum 6E 0E 5A 6C 7F 5B 8D 79 96 B0 CD B4 CF 85 DB 15E
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
```

```
FBF0 80 FF 10 28 C7 01 01 82 9A BA FE FF FF FC DB 124
FB00 1C 10 20 7B 0B 05 02 00 FF 71 79 F9 70 7F 14E
FB10 EF 95 85 05 4B 4D 51 21 25 ED 4F 7F 7F 15C
FB20 7F 4B 4B 4B 4B 4A 42 42 42 42 42 42 42 42 42 42
FB30 42 42 47 7F 7F 7F 7F 7F 00 FF 55 55 00 00 01 90 180
FB40 A1 BE A1 40 40 C0 C0 C0 C0 C0 C0 FF FF FF FF FF
FB50 00 F0 89 05 01 02 02 02 02 02 02 02 02 02 02 02
FB60 59 59 FF FC F9 F0 E0 FF 00 FF FF FF FF FF FF FF
FB70 11 21 21 21 41 49 97 E1 B1 83 45 B5 05 05 01 1F 186
FB80 00 7F 40 40 40 40 40 40 40 40 40 40 40 40 40 40
FB90 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F
FB00 15 0A 05 00 00 00 80 F0 F0 F0 F0 F0 F0 F0 F0 F0
FB10 87 04 7A 83 00 00 00 00 00 00 00 00 00 00 00
FB20 FB FC FC FC FF FF FC 7C 7F 01 C5 2B 2D 30 1C
FB30 49 89 13 0D 01 01 01 01 01 01 01 01 01 01 01
FB40 01 6A 55 6A 55 6A 5A 7F 7F 7F 7F 7F 7F 7F 7F
Sum 7C 90 A0 9E A1 D9 75 65 7B 4B CB B5 76 43 F3 0E 196
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FB50 5E 70 67 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F
FB60 66 50 AC 93 DC BC 03 00 00 80 FF FF FF FF FF
FB70 00 9E 4E 62 8B CE E5 24 29 25 05 0E 1E FF 20 14D
FB80 00 00 00 00 F0 FF FF FF 00 01 01 01 01 01 01 01
FB90 41 21 21 21 11 11 01 01 01 01 01 01 01 01 01
FA00 00 40 40 40 40 40 40 40 40 40 40 40 40 40 40
FA10 4A 7B 40 40 50 5B 5B 5B 5B 10 0B 0B CB 90 E0 1C3
FA20 00 80 40 80 1F F0 CB 20 20 10 11 12 0A 0A 0A 15
FA30 04 FF 00 7B 1E 09 09 09 07 03 01 01 02 42 41 EC
FA40 10 CB 34 02 01 00 00 82 FF 01 01 01 01 01 01
FA50 01 01 59 01 01 01 01 01 01 01 01 01 01 01 01
FA60 39 5C 4C 4C 4C 66 76 5E 4E 47 44 44 43 43 47 E0
FA70 4B 4B 46 41 41 40 40 40 7F 00 02 02 01 00 07 0E 16E
FA80 1F 1F 9F 8F 07 F1 8B 07 FB 7B 2B 24 24 DB 00 FF 17D
FA90 00 01 0C 03 EC 0B 08 20 F3 E4 FF FF FF FF FF
FA00 87 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Sum B5 EB FC 05 96 6C E5 1D F0 26 E6 E2 D6 A0 89 69 1EB
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FAF0 87 01 D9 25 A5 F9 FD FF FF FF FF FF FF FF FF
FB00 7F 00 40 40 40 40 40 40 40 40 40 40 40 40 40
FB10 45 45 45 45 45 45 47 4F 4C 4B FF 00 00 00 00 02 184
FB20 0C 7E 40 40 40 40 40 40 40 40 40 40 40 40 40
FB30 7F 10 10 0B 08 04 04 02 01 00 00 00 00 00 00 17A
FB40 00 00 00 00 00 00 00 00 00 00 FF 01 01 01 01 01
FB50 01 B1 71 0F 01 05 08 0B 08 0F 2F 1F 5F 3F 3F 182
FB60 7F 4A 4A 42 41 60 70 70 70 70 70 70 70 70 70
FB70 00 70 5A 00 00 00 00 00 00 00 00 00 00 00 00
FB80 0A 0A 0A 09 09 09 11 11 21 E1 47 B2 0A 00 FF 130
FB90 00 30 0B 04 1C 1E 7E 7E FE 7E 2B 21 01 21 39 1FA
FBA0 1F 0F 0F 07 07 03 03 00 00 00 00 00 00 00 00
FBB0 FF FF 0C 5C 5C 4C 44 47 44 5A 71 72 61 60 51 1C
FBC0 4E 62 72 64 4B 48 49 53 45 40 00 00 00 00 00 E0
FBD0 F2 62 64 08 05 69 91 00 00 0C 12 E1 BC 9E 4C 21 155
Sum AF 54 BA FE 7B 7F 3A B0 D9 27 1D B6 F5 70 76 93 1DD
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FBF0 40 FF 0C 0E 06 02 0C E7 64 24 1B 24 C7 18 7C 79 17E
FC00 42 B1 1A E4 41 4E 8B 09 84 FF 01 01 01 01 01 01
FC10 63 F5 09 85 89 91 01 0D 1D 01 21 12 C5 E5 180
FC20 C9 65 7C 5F 6B 7A 7F 7F 7F 7F 7F 7F 7F 7F 7F
FC30 40 40 40 40 60 7C 7F 7F 7F 20 12 80 FB 7F 8E 1E6
FC40 FF FF FF FF FF FF 7F 07 00 00 00 00 00 00 00
FC50 00 02 1C 2A 31 01 00 00 00 00 00 00 00 00 00
FC60 3F 00 00 00 00 00 00 00 00 00 11 69 D1 21 01 81 FF 1CB
FC70 1D 1D FF FF FF FF FF FF 7F 7F 01 01 01 01 01 01
FC80 00 7F 40 40 40 40 41 42 44 4B 4B 50 50 50 50 166
FC90 50 48 4B 4A 42 42 42 42 42 42 42 42 42 42 42
FDA0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
FDB0 3F 00 00 00 00 00 10 0B 08 02 02 01 01 01 01 143
FDC0 01 01 02 02 04 10 10 60 80 FF 01 01 01 01 01 01
FDD0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01
FDE0 01 40 40 43 5F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F
Sum 3D F8 0C BD 2E FA 76 9D E3 33 97 98 6E 6B CC 43 166
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FDF0 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F 7F
FD00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FD10 00 00 00 F0 FF FF FF FF FF FF FF FF FF FF FF
FD20 FF FF FF FF FF FF FF FF FF FF 01 01 01 01 01
FD30 FF FF FF FF FF FF FF FF FF FF 01 01 01 01 01
FD40 00 4E 7F 7F 7F 7F 7F 7C 7C 7C 7C 7C 7C 7C 7C
FD50 75 75 43 60 50 50 48 6A 6F FF FF FF FF 7F 87 192
FD60 97 40 0E 86 00 00 1F 7F 55 5F 0C 03 67 75 15 197
FD70 0F FF FC FF FF FF FF FF FF FF FF FF FF FF FF
FD80 01 00 00 00 00 00 00 00 00 00 01 01 01 01 01
FD90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FDA0 89 5A 47 40 60 76 7E 7E 7E 5F 7F 7F 7F 7F 110
FDB0 7F 4F 43 40 40 40 7F 7F 00 05 27 74 7E 7C 30 1E
FDC0 E4 07 0C FF FF FF FF FF FF FF FF FF FF FF FF
FDD0 00 00 63 73 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F
FDE0 FF FF 8F 8F 00 00 00 00 00 00 31 C1 01 F1 F9 FF 13A
Sum 7D 1F BD 5F EB 49 2D AB C4 97 5D 69 B5 EA 69 4B 10B
```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FD00 FF FF FF FF FD FD FD FD F9 F9 F1 E1 C1 01 01 FF 17B
FE00 00 7F 4A 4B 51 62 47 4F 5F 7F 4F 47 4F 4F 5F 114
FE10 5F 7F 7F 7F 7F 7E 7E 7E 7E 7F 5F 9F 0F 0F F3 150
FE20 FD FC FC FC FE FE FE F6 F2 F1 E3 EF 0A E1 C1 C1 1B8
FE30 41 FF FF FF FF FF FF FF GF GF GF GF GF GF GF GF 157
FE40 7E FF FF 8B 04 84 C2 42 E1 FF 91 E9 F5 F9 FD FD 1D2
FE50 FF FF FF FF BF 0F 0F 3F 1F 9F 0F 1F 1F 1F 0F 0F 110
FE60 0F 5C 4E 4E 4E 4E 60 60 60 60 60 60 60 60 60 1F4
FE70 4A 57 4B 4B 4B 4B 7F 7F 00 42 04 04 04 0C 95 198
FE80 4A 4B 22 C3 07 81 80 80 80 80 80 00 1F FF FF 115
FE90 60 11 6E 81 81 A1 91 0D 07 0B 0B 0B 0B 00 00 1B3
FEA0 06 0B 0C 7C 7E FF FF FF 00 BF BF 0F 07 07 07 1D0
FEB0 07 05 01 01 01 01 61 91 F1 9F 9F 9F FF FF 12C
FEC0 04 7E 7E 77 77 77 77 77 77 77 77 77 77 102
FED0 77 77 77 77 77 77 77 77 77 77 77 77 77 129
FEE0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF 1F0
Sum 39 05 91 7E 0D B5 DF 09 86 B8 59 99 68 38 DE 93 12B

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FEF0 FF FF FF FF 00 FF FF FF FF FF FF FF FF FF FF 1F1

```

```

FF00 FF FF FF FF FF FF FF FF FF FF FF FF FF 11 .. 16 17 18 19 1A 1B 1C 1D 1E 1F 1A
FF10 FB FB FB FB FB FB FB FB FB FB FB FB FB FB FB 1B0
FF20 FB 77 77 77 77 77 77 77 77 77 77 77 77 77 77 1F4
FF30 77 77 77 77 77 77 77 77 77 77 77 77 77 77 1BA
FF40 FF FF FF FF FF FF FF FF FF FF FF FF FF FF 1F1
FF50 00 FF FF FF FF FF FF FF FF FF FF FF FF FF 1F1
FF60 FF FF FF FF FF FF FF FF FF FF FF FF FF FF 1D6
FF70 FB FB FB FB FB FB FB FB FB FB FB FB FB FB 1C0
FF80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1F9
FF90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
FFA0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
FFB0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
FFC0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
FFD0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
FFE0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 1F8
Sum 5E DF DB 0E DB D9 0E E7 62 63 5C 67 5B 6B 5C 63 174

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 15um
FFF0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
Sum FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8

```

グラフィック・花札プログラム・リスト

```

10 REM *****
20 REM *   クラシック ゲーム   *
30 REM *   コイ コイ   *
40 REM *   REV 1.0   *
50 REM *****
70 CLR
90 GRAPHIC,01:PRINTCHR$(6)
100 DIM TA(6,2),F(40,3),F(4,26),F1(48),F2(23),F3(11),F1$(49),F2$(49),T1$(30)
120 DEF FNZ(X)=X-INT(X/12)*12
200 REM
240 DATA 0,0,0,0,0,0
250 DATA 5,4,3,3,3,3
260 DATA 5,4,3,3,3,3
300 REM 79 * GRAPH 描き
310 P=55001:A$="" :FOR X=P TO P+191:A$=A$+CHR$(PEEK(X)):NEXT:T1$(1)=A$
315 P=55193:A$="" :FOR X=P TO P+95:A$=A$+CHR$(PEEK(X)):NEXT:T1$(2)=A$
320 P=54001:A$="" :FOR X=P TO P+239:A$=A$+CHR$(PEEK(X)):NEXT:T1$(29)=A$
322 P=54241:A$="" :FOR X=P TO P+239:A$=A$+CHR$(PEEK(X)):NEXT:T1$(30)=A$
325 GOSUB9000
330 P=56001
335 PRINT"サマ イマ ー ラ ヲ コンテ イマス."
340 FOR GR=1 TO 49
350 P=PD+192*(GR-1)
360 A$="" :FOR X=P TO P+95:A$=A$+CHR$(PEEK(X)):NEXT:F1$(GR)=A$
370 A$="" :FOR X=P+96 TO P+191:A$=A$+CHR$(PEEK(X)):NEXT:F2$(GR)=A$
380 PRINT49-GR:NEXT
385 CONSOLE:GOSUB9100:GOSUBB600:GOSUBB600
390 A$="" :FOR I=0 TO 95:A$=A$+CHR$(0):NEXT:F1$(0)=A$:F2$(0)=A$
420 P=55289:FOR GR=3 TO 6:P=PD+96*(GR-3)
430 A$="" :FOR X=P TO P+95:A$=A$+CHR$(PEEK(X)):NEXT:T1$(GR)=A$
440 NEXT
445 GOSUB9130
450 P=55673:FOR GR=7 TO 14:P=PD+16*(GR-7)
460 A$="" :FOR X=P TO P+15:A$=A$+CHR$(PEEK(X)):NEXT:T1$(GR)=A$
470 NEXT
480 A$=CHR$(0)+CHR$(0)+B$="" :FOR I=1 TO 48:B$=B$+A$:NEXT:T1$(15)=B$
490 A$=CHR$(0)+CHR$(0)+B$="" :FOR I=1 TO 112:B$=B$+A$:NEXT:T1$(16)=B$
510 P=55801:FOR GR=18 TO 27:P=PD+16*(GR-18)
520 A$="" :FOR X=P TO P+15:A$=A$+CHR$(PEEK(X)):NEXT:T1$(GR)=A$
530 NEXT
540 P=55961:A$="" :FOR X=P TO P+31:A$=A$+CHR$(PEEK(X)):NEXT:T1$(28)=A$
600 CURSOR0,20:PRINT"79 * カマサセ ムタメ ショウ ラフ シテ 2000Key   ラ オンテ 79 * ヴィ."
610 GET A$:A=RND(1):IF A$=" " THEN610
800 REM
810 CONSOLEC80
820 OYA=INT(RND(1)*2)
900 REM TITLE ヒヤッ
910 PRINTCHR$(6):GRAPHIC 01
915 ST=ST+1:X=232:Y=B:PF=PF+1:GOSUBB510
920 POSITION10,0:PATTERN -24,T1$(1):PATTERN -24,T1$(2)
930 POSITION4,8:PATTERN -16,T1$(3):POSITION4,72:PATTERN -16,T1$(4)
940 FOR I=0 TO 7:POSITION 18+4,184:PATTERN -8,T1$(1+7):NEXT
1020 REM
1030 FOR I=1 TO 48:F1(I)=I:NEXT
1035 RESTORE:FOR Y=0 TO 2
1036 FOR X=1 TO 6:READ TA(X,Y):NEXT X
1037 NEXT Y
1050 IF OYA=0 THEN 1100
1060 K=0:GOSUB1210
1070 K=1:GOSUB1210
1080 K=2:GOSUB1210
1090 GOTO1140
1100 K=1:GOSUB1210
1110 K=0:GOSUB1210
1120 K=2:GOSUB1210

```

プログラム・リスト

```

1140 A=0:FOR I=1 TO 48:IF F1(I)=0 THEN1160
1150 F2(A)=F1(I):A=A+1
1160 NEXT
1170 FOR I=0 TO 50:A=INT(RND(1)*24):B=INT(RND(1)*24)
1180 C=F2(A):F2(A)=F2(B):F2(B)=C:NEXT
1190 GOTO1300
1200 REM
1210 FOR J=0 TO 11:F3(J)=0:NEXT
1220 FOR I=1 TO 8
1230 A=INT(RND(1)*48)+1:IF F1(A)=0 THEN1230
1240 B=FNZ(A):F3(B)=F3(B)+1:IF F3(B)>2 THEN F3(B)=F3(B)-1:GOTO1230
1250 F(K,I)=A:F1(A)=0
1260 NEXT:RETURN
1300 FOR I=0 TO 2:X=I:GOSUB6010:NEXT
1320 REM ヲカ カメ
1330 KAI=0:HM=8:Z=8:GOSUB7090:PB=8:GOSUB7170
1340 POSITION224,144:PATTERN -24,F1*(49):POSITION224,168:PATTERN -24,F2*(49)
1350 X=56:Y=8:PP=0:GOSUB6510:POSITION80,7:PATTERN -16,T1*(28)
1360 X=56:Y=72:PP=1:GOSUB6510:POSITION80,71:PATTERN -16,T1*(28)
1500 PC=0:PM=0:HC=8:SC=0:SM=0
1505 FOR K=3 TO 4:FOR I=0 TO 24:F(K,I)=0:NEXT:NEXT
1508 TA(0,0)=9:TA(0,1)=9:TA(0,2)=9
1510 IF OYA=1 THEN1800
1520 KAI=KAI+1:IF HC=0 THEN8110
1530 GOSUB5010:XC=XA:XD=XB
1540 IF B=0 THEN PB=PB+1:F(2,PB)=F(0,1):F(0,1)=99:GOTO1570
1550 F(3,PC+1)=C:F(3,PC+2)=B:PC=PC+2:X=3:FL=PC:GOSUB6020
1560 F(2,CQ(J,1))=99:X=2:FL=PB:GOSUB6020:PB=PB-1
1565 F(0,CQ(J,0))=99
1570 X=0:FL=HC:GOSUB6020:HC=HC-1:F=3:Z=PC:GOSUB7010:Z=PB:GOSUB7170:REM カメシ
1575 GOSUB6600
1580 F=0:GOSUB6110
1590 IF B=0 THEN PB=PB+1:F(2,PB)=C:X=2:FL=PB:GOSUB6020:GOTO1620
1600 F(3,PC+1)=C:F(3,PC+2)=B:PC=PC+2:X=3:FL=PC:GOSUB6020
1610 F(2,CQ(J,1))=99:X=2:FL=PB:GOSUB6020:PB=PB-1
1620 F=3:Z=PC:GOSUB7010:Z=PB:GOSUB7170:REM ヲカメ
1630 IF (TA(XA,1)=0)*(TA(XA,0)<4) THEN TA(XA,0)=4:GOTO1690
1635 IF (TA(XB,1)=0)*(TA(XB,0)<4) THEN TA(XB,0)=4:GOTO1690
1640 IF (TA(XC,1)=0)*(TA(XC,0)<4) THEN TA(XC,0)=4:GOTO1690
1645 IF (TA(XD,1)=0)*(TA(XD,0)<4) THEN TA(XD,0)=4:GOTO1690
1650 GOTO1800
1690 IF SM=1 THEN A=4:GOTO7510
1700 FOR XT=0 TO 5
1710 IF (TA(XT,2)=1)*(TA(XT,0)=2) THEN A=4:GOTO7510
1720 NEXT
1730 SC=1:A=4:GOSUB6010
1800 KAI=KAI+1:IF HM=0 THEN8110
1805 CURSOR0,24:INPUT"トレヲキリヌカフ" :S
1810 CURSOR0,23:PRINT" " :IF S>HM THEN1805
1820 C=F(1,S):GOSUB6510:XC=XA:XD=XB
1900 F(1,S)=99:X=1:FL=HM:GOSUB6020:HM=HM-1:GOSUB7080
1910 POSITION 0,184:PATTERN -8,T1*(16):FOR I=1 TO HM:POSITION I*32-28,184:PATTE
RN -8,T1*(I+6):NEXT
1920 F=4:Z=PM:GOSUB7010:Z=PB:GOSUB7170
1930 C=F2(KAI-1):POSITION224,144:PATTERN -24,F1*(C):POSITION224,168:PATTERN -24
,F2*(C)
1935 GOSUB6600
1940 GOSUB6510
2000 IF (TA(XA,2)=0)*(TA(XA,0)<4) THEN TA(XA,0)=5:GOTO2020
2005 IF (TA(XB,2)=0)*(TA(XB,0)<4) THEN TA(XB,0)=5:GOTO2020
2010 IF (TA(XC,2)=0)*(TA(XC,0)<4) THEN TA(XC,0)=5:GOTO2020
2012 IF (TA(XD,2)=0)*(TA(XD,0)<4) THEN TA(XD,0)=5:GOTO2020
2015 GOTO1520
2020 CURSOR 0,24:INPUT"ヨウツ" = 0 ,コイイ = 1 ? " :SM
2030 CURSOR 0,23:PRINT" "
2040 IF SM=0 THEN A=5:GOTO7510
2090 IF SM=1 THEN SC=0:A=5:GOSUB6010:GOTO1520
2060 GOTO 2020
5000 REM
5010 CC=0:CQ(0,0)=0
5020 FOR JO=1 TO HC
5030 C=F(0,JO)
5040 FOR J2=1 TO PB
5050 B=F(2,J2):IF FNZ(B)=FNZ(C) THEN GOSUB5510
5060 NEXT
5070 NEXT
5080 GOSUB5710
5090 RETURN
5500 REM
5510 IF (B<28)+(B>33)*(B<39)+(B=42)+(B=43) THEN GOTO5600
5520 CC=CC+1:CQ(CC,2)=0:CQ(CC,3)=0
5530 IF B=48 THEN A=1:GOTO5580
5540 IF B>43 THEN A=3:GOTO5580
5550 IF (B>38)*(B<42) THEN A=6:GOTO5580
5560 IF (B>30)*(B<34) THEN A=5:GOTO5580

```

```

5570 IF (B>27)*(B<31) THEN A=4
5580 CO(CC,2)=A:GOTO5620
5600 IF (C<28)+((C<33)*(C<39))+(C=42)+(C=43) THEN CO(0,0)=JO:CO(0,1)=J2:RETURN
5610 CC=CC+1:CO(CC,2)=0:CO(CC,3)=0
5620 IF C=48 THEN A=1:GOTO5670
5630 IF C=43 THEN A=3:GOTO5670
5640 IF (C>38)*(C<42) THEN A=6:GOTO5670
5650 IF (C>30)*(C<34) THEN A=5:GOTO5670
5660 IF (C>27)*(C<31) THEN A=4:GOTO5670
5665 GOTO5680
5670 CO(CC,3)=A
5680 CO(CC,0)=JO:CO(CC,1)=J2:RETURN
5700 REM
5710 IF CC=0 THEN5900
5715 FOR I=1 TO 2
5720 FOR J=1 TO CC
5725 XA=CO(J,2):XB=CO(J,3):T1=TA(XA,0):T2=TA(XB,0):T3=TA(XA,1):T4=TA(XB,1):T5=TA(XA,2):T6=TA(XB,2)
5730 IF (T3>1)*(T4>1) THEN5745
5735 IF ((XA=3)*(T5=2))+((XB=3)*(T6=2)) THEN 5825
5740 IF (T1=1)*(T2=1) THEN5825
5745 NEXT
5750 FOR J=1 TO CC
5755 XA=CO(J,2):XB=CO(J,3):T1=TA(XA,0):T2=TA(XB,0):T3=TA(XA,1):T4=TA(XB,1):T5=TA(XA,2):T6=TA(XB,2)
5760 IF (T5>1)*(T6>1) THEN5775
5765 IF ((XA=3)*(T3=2))+((XB=3)*(T4=2)) THEN 5825
5770 IF (T1=2)*(T2=2) THEN5825
5775 NEXT:NEXT
5780 FOR J=1 TO CC
5785 XA=CO(J,2):XB=CO(J,3):T1=TA(XA,0):T2=TA(XB,0)
5790 IF (T1=0)*(T2=0) THEN5825
5795 NEXT
5800 FOR J=1 TO CC
5805 XA=CO(J,2):XB=CO(J,3):T1=TA(XA,0):T2=TA(XB,0)
5810 IF (T1=0)+(T2=0) THEN5825
5815 NEXT
5820 J=1:XA=CO(J,2):XB=CO(J,3):T1=TA(XA,0):T2=TA(XB,0)
5825 IF XA=0 THEN5855
5830 IF TA(XA,1)<>0 THEN5840
5835 XA=2:TA(XA,1)=1:IF TA(1,1)=4 THEN XA=1:TA(XA,1)=1
5840 TA(XA,1)=TA(XA,1)-1
5845 IF T1=0 THEN TA(XA,0)=1
5850 IF T1=2 THEN TA(XA,0)=3
5855 IF XB=0 THEN5855
5860 IF TA(XB,1)<>0 THEN5870
5865 XB=2:TA(XB,1)=1:IF TA(1,1)=4 THEN XB=1:TA(XB,1)=1
5870 TA(XB,1)=TA(XB,1)-1
5875 IF T2=0 THEN TA(XB,0)=1
5880 IF T2=2 THEN TA(XB,0)=3
5885 B=F(2,CO(J,1)):C=F(0,CO(J,0))
5890 RETURN
5900 IF CO(0,0)=0 THEN B=0:RETURN :REM NO MATCH
5910 J=0:GOTO5885
6000 REM
6010 FL=B
6020 IF FL<=1 THEN RETURN
6030 MX=FL-1:FL=0
6040 FOR JJ=1 TO MX
6050 IF F(X,JJ)*F(X,JJ+1) THEN A=F(X,JJ):F(X,JJ)=F(X,JJ+1):F(X,JJ+1)=A:FL=JJ
6060 NEXT
6070 GOTO6020
6100 REM
6110 C=F(2,KAI-1):POSITION224,144:PATTERN -24,F1*(C):POSITION224,168:PATTERN -24,F2*(C):REM 79" ヒヤ79"
6120 CC=0:CO(0,0)=0:JO=1
6125 IF PB=0 THEN6160
6130 FOR J2=1 TO PB
6140 B=F(2,J2):IF FNZ(B)=FNZ(C) THEN GOSUB5510
6150 NEXT
6160 GOSUB5710:C=F(2,KAI-1)
6170 GOSUBB600:GOSUBB600
6180 POSITION224,144:PATTERN -24,F1*(49):POSITION224,168:PATTERN -24,F2*(49)
6190 RETURN
6500 REM
6510 CC=0:CO(0,0)=0
6515 IF PB=0 THEN6550
6520 FOR J2=1 TO PB
6530 B=F(2,J2):IF FNZ(B)=FNZ(C) THEN GOSUB5510
6540 NEXT
6550 IF CC<>0 THEN6580
6560 IF CO(0,0)=0 THEN XA=0:XB=0:PB=PB+1:F(2,PB)=C:X=2:FL=PB:GOSUB6020:GOTO6760
6570 J=0:B=F(2,CO(J,1)):XA=CO(0,2):XB=CO(0,3):GOTO6740
6580 IF (CC=1)+(CO(1,1)=CO(2,1)) THEN J=1:B=F(2,CO(1,1)):GOTO6690
6590 CURSOR0,24:INPUT"マシン79" 79"ヒヤ79" "
6600 CURSOR0,23:PRINT" "

```

```

6610 IF A=1 THEN MN=1:MX=23:MM=35:GOTO6660
6620 IF A=5 THEN MN=24:MX=35:MM=0:GOTO6660
6630 IF A=10 THEN MN=34:MX=43:MM=0:GOTO6660
6640 IF A=20 THEN MN=44:MX=48:MM=0:GOTO6660
6650 GOTO6690
6660 FOR J=1 TO CC
6670 B=F(2,CD(J,1)):IF (B>=MN)*(B<=MX)+(B=MM) THEN6690
6680 NEXT:GOTO6690
6690 XA=CD(J,2):XB=CD(J,3)
6695 IF XA=0 THEN6720
6700 IF TA(XA,2)<0 THEN6710
6705 XA=2:TA(XA,2)=1:IF TA(1,2)=4 THEN XA=1:TA(XA,2)=1
6710 TA(XA,2)=TA(XA,2)-1
6715 IF TA(XA,0)<2 THEN TA(XA,0)=TA(XA,0)+2
6720 IF XB=0 THEN6745
6725 IF TA(XB,2)<0 THEN6735
6730 XB=2:TA(XB,2)=1:IF TA(1,2)=4 THEN XB=1:TA(XB,2)=1
6735 TA(XB,2)=TA(XB,2)-1
6740 IF TA(XB,0)<2 THEN TA(XB,0)=TA(XB,0)+2
6745 PM=PM+1:F(4,FM)=B:PM=PM+1:F(4,FM)=C:X=4:FL=PM:GOSUB6020:REM PMY-I
6750 F(2,CD(J,1))=99:Y=2:FL=PB:GOSUB6020:PB=PB-1
6760 GOSUB8600:GOSUB8600
6770 POSITION224,144:PATTERN -24,F1*(49):POSITION224,168:PATTERN -24,F2*(49)
6780 RETURN
7000 REM
7010 DY=64*(F-3)+24:FOR X=0 TO 7:POSITION X*32,DY:PATTERN -24,F1*(0):POSITION X
*32,DY+24:PATTERN -24,F2*(0):NEXT
7020 IF Z=0 THEN RETURN
7030 IF Z<9 THEN X1=32:GOTO7050
7040 X1=224/(Z-1)
7050 FOR X=1 TO Z:I=F(X,X)
7060 POSITION X1*(X-1),DY:PATTERN -24,F1*(I):POSITION X1*(X-1),DY+24:PATTERN -2
4,F2*(I)
7070 NEXT:RETURN
7080 REM
7090 FOR X=0 TO 6:POSITION X*32,136:PATTERN -24,F1*(0):POSITION X*32,160:PATTER
N -24,F2*(0):NEXT
7100 IF MM=0 THEN RETURN
7110 IF MM=B THEN X1=27:GOTO7130
7120 X1=32
7130 FOR X=1 TO MM:I=F(1,X):IF I=0 THEN7150
7140 POSITION X1*(X-1),136:PATTERN -24,F1*(I):POSITION X1*(X-1),160:PATTERN -24
,F2*(I)
7150 NEXT:RETURN
7160 REM
7170 FOR DY=0 TO 7:FOR DX=0 TO 1:POSITION 256+DX*32,DY*24:PATTERN -24,F1*(0):NE
XT:NEXT
7180 IF Z=0 THEN RETURN
7210 Z1=INT(Z/2):Z2=Z-Z1
7220 IF Z1=0 THEN7275
7230 IF Z1=1 THEN Y1=48:GOTO7250
7240 Y1=144/(Z1-1):IF Y1>48 THEN Y1=48
7250 FOR Y=1 TO Z1:I=F(2,Y+Z1)
7260 POSITION 256,Y1*(Y-1):PATTERN -24,F1*(I):POSITION 256,Y1*(Y-1)+24:PATTERN
-24,F2*(I)
7270 NEXT
7275 IF Z2=1 THEN Y2=48:GOTO7290
7280 Y2=144/(Z2-1):IF Y2>48 THEN Y2=48
7290 FOR Y=1 TO Z2:I=F(2,Y+Z1)
7300 POSITION 288,Y2*(Y-1):PATTERN -24,F1*(I):POSITION 288,Y2*(Y-1)+24:PATTERN
-24,F2*(I)
7310 NEXT:RETURN
7500 REM 7527 7577
7510 M1#="A3GEGA6B1RIA3GEGD6R3E5G1RGRE2R0E2R0D3C":M2#=""DEDC-A7":M3#=""D4R1-A4R1-
A7":TEMPO:MUSIC M1:MUSIC M2:MUSIC M1:MUSIC M3
7515 FOR I=0 TO 3
7520 B=(A(4)+8)*8:POSITION4,B:PATTERN -16,T1*(15):POSITION4,B:PATTERN -16,T1*(3
)
7530 M#=""C3":MUSIC M#:FOR W=0 TO 20:C=3*2:NEXT
7540 POSITION4,B:PATTERN -16,T1*(15):POSITION4,B:PATTERN -16,T1*(6)
7550 M#=""G7":MUSIC M#:FOR W=0 TO 20:C=3*2:NEXT
7560 NEXT
7570 FOR DY=0 TO 7:FOR DX=0 TO 1:POSITION 256+DX*32,DY*24:PATTERN -24,F1*(0):NE
XT:NEXT
7580 CURSOR 65,1:PRINT "7577 7577":DY=3:P=0
7610 IF TA(1,0)=A THEN P=F+15:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2:GOTO76
25
7615 IF TA(2,0)=A THEN P=F+10:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2:GOTO76
25
7620 IF TA(3,0)=A THEN P=F+8:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2
7625 IF TA(4,0)=A THEN P=F+6:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2
7630 IF TA(5,0)=A THEN P=F+5:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2
7635 IF TA(6,0)=A THEN P=F+5:CURSOR 65,DY:PRINT "7577 7577 7577 7577":DY=DY+2
7640 IF A=4 THEN P=F+P+1:GOTO7660
7650 P=P+1:P=

```

```

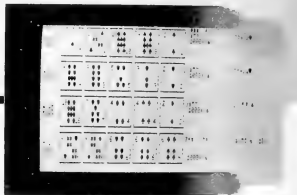
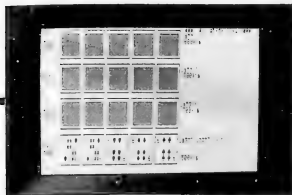
7660 X=56:Y=8:PP=P0:GOSUBB510:POSITIONB0,7:PATTERN -16,T1*(28)
7670 X=56:Y=72:PP=P1:GOSUBB510:POSITIONB0,71:PATTERN -16,T1*(28)
7672 IF ST=12 THEN CURSOR0,20:PRINT"シュウワウ ティス。":END
7675 CURSOR0,24:PRINT"Push Any Key":
7676 GET A$:IF A$="" THEN7676
7680 IF A=4 THEN OYA=0:GOTO910
7690 IF A=5 THEN OYA=1:GOTO910
8000 REM コイイ ヒョウ
8010 M$="C3RERGEGRGDCDEGRESG+C3+E+D+CGR16A3G+C1R+CR+CR2":TEMPO6:MUSIC M$
8015 FOR I=0 TO 3
8020 B=(A-4)*64+B:POSITION4,B:PATTERN -16,T1*(15):POSITION4,B:PATTERN -16,T1*(A
-1)
8030 M$="+C3":MUSIC M$:FOR W=0 TO 20:C=3^2:NEXT
8040 POSITION4,B:PATTERN -16,T1*(15):POSITION4,B:PATTERN -16,T1*(5)
8050 M$="G3":MUSIC M$:FOR W=0 TO 20:C=3^2:NEXT
8060 NEXT:RETURN
8100 REM テマ テル
8110 IF SC=1 THEN A=4:GOTO7570
8120 IF SM=1 THEN A=5:GOTO7570
8130 IF OYA=0 THEN P=6:P0=P0+P:A=5:GOTO7660
8140 P=6:P1=P1+P:A=4:GOTO7660
8500 REM トワシ ヒョウ Y=8,72 PP=P0,P1
8510 T1$="":B$=CHR$(0)+CHR$(0):FOR I=1 TO 24:T1$=T1$+B$:NEXT
8520 POSITION X,Y:PATTERN -16,T1$
8530 L=LEN(STR$(PP)):FOR I=0 TO 2:J=L-I:IF J<=0 THEN 8550
8540 K$=MID$(STR$(PP),J,1):K$=VAL(K$):POSITION X+16-I*8,Y:PATTERN -16,T1*(18+K)
8550 NEXT
8560 RETURN
8600 FOR W=0 TO 100:GET W$:IF W$="C" THEN CONSOLE:COPY/P 2:CONSOLE
8610 NEXT
8620 RETURN
9000 REM エンメイ
9010 PRINTCHR$(6):CONSOLEC40:GRAPH01
9011 POSITION68,0:PATTERN -24,T1*(1):PATTERN -24,T1*(2):PATTERN -24,T1*(2)
9015 POSITION136,30:PATTERN -24,T1*(29)
9020 CURSOR 0,8:PRINT"ハッパ (ハ)カハハハ) ト オシシ ヨウワウ ティ イフク マ アツヒ イキマカ トワシ
ノ テンワウ ト フルハハハハ) ティ ティス。"
9030 PRINT:PRINT"「フク」カ「ティ」ハ「コイイ」マハ「ショウワウ」ヲ エンシシ シマス。「コイイ」ハ
「ム」ヲ ツツクハ「ショウワウ」ハ ヲコシ「ヒヤシ」チス。"
9040 PRINT:PRINT"フクハ「コイイ」ヲ エンシシ スルハ フクハ「フク」ヲ ツツクハハ「オハコト」カ「チ」
ツツク。"
9045 POSITION224,168:PATTERN -24,T1*(30)
9050 PRINT:PRINT"Push Any Key"
9060 GET A$:IF A$="" THEN9060
9070 RETURN
9100 PRINTCHR$(6):CURSOR0,3:PRINT"ゴゴゴ":FOR LL=44 TO 48:POSITION64+(LL-44)*32,0
:PATTERN -24,F1*(LL):POSITION64+(LL-44)*32,24:PATTERN -24,F2*(LL):NEXT
9110 CURSOR0,9:PRINT"ゴゴゴ":FOR LL=44 TO 47:POSITION64+(LL-44)*32,48:PATTERN -24,
F1*(LL):POSITION64+(LL-44)*32,72:PATTERN -24,F2*(LL):NEXT
9120 CURSOR0,15:PRINT"シシシ シシ ノ フク ノ シマイ":RETURN
9130 PRINTCHR$(6):GRAPH0:CURSOR0,3:PRINT"フク":FOR LL=28 TO 30:POSITION64+(LL-2
8)*32,0:PATTERN -24,F1*(LL):POSITION64+(LL-28)*32,24:PATTERN -24,F2*(LL):NEXT
9140 CURSOR0,9:PRINT"フク":FOR LL=31 TO 33:POSITION64+(LL-31)*32,48:PATTERN -24
,F1*(LL):POSITION64+(LL-31)*32,72:PATTERN -24,F2*(LL):NEXT
9150 CURSOR0,15:PRINT"シシシシシハハハハ":FOR LL=39 TO 41:POSITION64+(LL-39)*32,96:PATTERN
-24,F1*(LL):POSITION64+(LL-39)*32,120:PATTERN -24,F2*(LL):NEXT:RETURN
9900 CONSOLE:FOR I=0 TO 1:GRAPH0
9910 FOR Y=0 TO 3:FOR X=0 TO 5
9920 POSITION X*50,Y*50:PATTERN -24,F1*(Y*12+X+1+I*6)
9930 POSITION X*50,Y*50+24:PATTERN -24,F2*(Y*12+X+1+I*6)
9940 NEXT:NEXT
9950 COPY/P 2
9960 NEXT
9970 STOP

```

4人ポーカー



■大場啓二



おなじみのポーカーゲームですが、より人間くさいものにするためにランダム関数をおおに利用しました。3人をコンピュータが受け持ちます。もちろん、ポーカーフェイスが使えます。ですから、いままでのポーカーゲームとは一味違ったものになっていると思います。なお、G-RAMは必要ありません。

★遊び方

RUNさせると、始めにタイトルがでます。そして、どれかキーを押すと何回対戦するか聞いてくるので、1～10の範囲で答えてください。

ランダムに親を決め、その次の人からカードを配ります。カードの交換は一枚ずつ「カエルカ」と聞いてくるので、YかNで答えてください。次に、勝負するか聞いてくるのでYかNで同様に答えてください。掛け金は1,000ドルをリミットとして、少しずつ上げていきます。そして1,000ドルになるまで何回も聞いてきます。なお、掛け金は（現在の掛け金）+ [押した数] × 100ドルなので注意してください。とにかく、実際に遊んでみてください。

★プログラムについて

A(I, J) …… カードのデータ (16～20=あなたのカード、I=図柄、J=数)

B(X, S) …… 役のデータ

タイトルやカードの表示はすべてストリング処理関数で

行なっています。この関数も使い方を工夫すると結構、面白いことができるので、みなさんも多に活用しましょう。

BASICによるプログラムなので理解し易いと思いますが、一部、説明をしておきます。

- 100～160 …… カードの並べ替え。
- 1000～1260 …… カード出力のデータ。
- 4000～4090 …… 「勝負」か「降りる」かの判断。
- 5200 …… カードの替え方を決める。
- 6000～6600 …… 相手の判断。
- 6700 …… オープンして、勝負を判定。配当金を求める。
- 9000 …… 表紙1
- 9200 …… 表紙2
- 10000 …… 総合順位の決定（成績発表）

★最後に

全体に渡って、行の長さが比較的長いものになっているので、プログラムの入力はCONSOLE C80, Rの状態で行なった方がやり易いと思います。また、他のマイコンへの移植は容易にできると思います。

とにかく、MZBはBASICの処理速度が速く、メモリも気にしないで使えるので助かりました。




```

1 REM 4人ボーカールゲーム by KEIZI. OBA. 1981/12/21
2 TEMPO 7:PRINTCHR$(6): GOTO9000
5 CLR
12 DIMA(4):A(1)=5000:A(2)=A(1):A(3)=A(1):A(4)=A(1):K1=1
15 TEMPO 7:GOSUB400
20 DIM A(4,15)
25 CONSOLEC80,R
30 FOR I=0 TO 4:FOR J=0 TO 15
40 A(I,J)=0: NEXT J:NEXT I
50 PRINTCHR$(6)
55 GOSUB5000:GOTO 450
60 P=1:GOSUB 200:Y=0:GOSUB2100:RETURN
70 P=6:GOSUB 200:Y=6:GOSUB2100:RETURN
80 P=11:GOSUB 200:Y=12:GOSUB2100:RETURN
90 P=16:GOSUB 200:Y=18:GOSUB2000:RETURN
100 DIM U(4):U(1)=1:U(2)=6:U(3)=11:U(4)=16
110 FOR J=14 TO 2 STEP -1:FOR I=1 TO 4
120 IF A(I,J)=1 THEN A(I,J)=U(1):U(1)=U(1)+1
130 IF A(I,J)=6 THEN A(I,J)=U(2):U(2)=U(2)+1
140 IF A(I,J)=11 THEN A(I,J)=U(3):U(3)=U(3)+1
150 IF A(I,J)=16 THEN A(I,J)=U(4):U(4)=U(4)+1:GOSUB170
160 NEXT I:NEXT J
165 DIME(R):FOR R=1 TO 5:R=0:NEXT R:GOTO 5040
170 Y=18:X=(U(4)-17)*9+7:GOSUB2002
180 RETURN
200 C=0
210 FOR I=1 TO 4:FOR J=2 TO 14
220 IF A(I,J)=0 THEN C=C+1
230 NEXT J:NEXT I
240 IF C>0 THEN 290
250 FOR I=1 TO 4:J=2 TO 14
260 IF A(I,J)=21 THEN A(I,J)=0
270 NEXT J:NEXT I
290 =INT(4*RND(1)+1):J=INT(13*RND(2)+2)
300 IF A(I,J)<>0 THEN 290
310 A(I,J)=P:RETURN
400 DIM P(1):P(1)=INT(RND(3)*4+1):CONSOLEC40,N:CURSOR15,15:ONP(1)GOSUB410,420,43
0,440:CURSOR15,8:PRINTK1;" MUSE" MUSIC" CDEFGAB+CR+CBAGFEDC":RETURN
410 PRINT"PA" MZ-K":RETURN
420 PRINT"PA" MZ-C":RETURN
430 PRINT"PA" MZ-B":RETURN
440 PRINT"PA" 777":RETURN
450 ON P(1)GOSUB 470,480,490,460:GOTO100
460 FORR=1 TO5:FORL=1TO4:ONLGOSUB60,70,80,90:NEXTL:NEXTR:RETURN
470 FORR=1 TO5:FORL=1TO4:ONLGOSUB70,80,90,60:NEXTL:NEXTR:RETURN
480 FORR=1 TO5:FORL=1TO4:ONLGOSUB80,90,60,70:NEXTL:NEXTR:RETURN
490 FORR=1 TO5:FORL=1TO4:ONLGOSUB90,60,70,80:NEXTL:NEXTR:RETURN
1000 PRINTCHR$(151)+STRING$(CHR$(155),7)+CHR$(149)
1010 FORT=1TO4:PRINT STRING$(CHR$(3),X)+CHR$(154)+SPACE$(7)+CHR$(154)+NEXTT
1020 PRINTSTRING$(CHR$(3),X)+CHR$(152)+STRING$(CHR$(155),7)+CHR$(150):RETURN
1030 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(49):PRINT:PRINT STRING$(CHR$(3),4+X)+
CHR$(0)+PRINTSTRING$(CHR$(3),7+X)+CHR$(49):REM 1:RETURN
1040 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(50)+STRING$(CHR$(3),2)+CHR$(0):PRINT:
PRINT:PRINTSTRING$(CHR$(3),4+X)+CHR$(0)+CHR$(3)+CHR$(3)+CHR$(50):REM2:RETURN
1050 PRINT:PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(51)+STRING$(CHR$(3),2)+CHR$(0):PRINT:
PRINTSTRING$(CHR$(3),4+X)+CHR$(0)
1060 PRINTSTRING$(CHR$(3),4+X)+CHR$(0)+CHR$(3)+CHR$(3)+CHR$(51):REM3:RETURN
1070 PRINT:PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(52):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NT:PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):" " :CHR$(52):REM4:RETURN
1080 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(53):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NTSTRING$(CHR$(3),4+X)+CHR$(0)
1090 PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):" " :CHR$(53):REM5:RETURN
1100 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(54):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0)
1110 PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):" " :CHR$(54):REM6:RETURN
1120 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(55):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NTSTRING$(CHR$(3),4+X)+CHR$(0):PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0)
1130 PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):" " :CHR$(55):REM7:RETURN
1140 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(56):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):NEXTT
1150 PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0):" " :CHR$(56):REM8:RETURN
1160 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(57):" " :CHR$(0):" " :CHR$(0):PRINT:PRI
NTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0)
1170 PRINTSTRING$(CHR$(3),3+X)+STRING$(CHR$(0),3):PRINTSTRING$(CHR$(3),3+X)+CHR
$(0):" " :CHR$(0):" " :CHR$(57):REM9:RETURN
1180 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(49)+CHR$(48)+CHR$(0):" " :CHR$(0):FOR
T=1TO2:PRINTSTRING$(CHR$(3),3+X)+CHR$(0)+CHR$(0)+CHR$(0):NEXTT
1190 PRINTSTRING$(CHR$(3),3+X)+CHR$(0):" " :CHR$(0)+CHR$(49)+CHR$(48):REM10:RETU
RN
1200 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(74):" " :CHR$(36)+CHR$(36):" " :CHR$(0)
+PRINTSTRING$(CHR$(3),3+X)+CHR$(35)+CHR$(35)+CHR$(47)
1210 PRINTSTRING$(CHR$(3),4+X)+CHR$(47)+CHR$(35)+CHR$(35):PRINTSTRING$(CHR$(3),
2+X)+CHR$(0)+CHR$(47):" " :CHR$(36)+CHR$(36)+CHR$(74):REM11:RETURN
1220 PRINT:PRINTSTRING$(CHR$(3),1+X)+CHR$(81):" " :CHR$(37)+CHR$(37):" " :CHR$(0)
+PRINTSTRING$(CHR$(3),3+X)+CHR$(35)+CHR$(35)+CHR$(47)

```



```

5020 E$="" " :CURSOR0,3:PRINTE$:CURSOR0,9:PRINTE$:CURSOR0,15:PRINTE$:CURSOR
0,21:PRINTE$
5030 CURSOR0,3:PRINTA(1):CURSOR0,9:PRINTA(2):CURSOR0,15:PRINTA(3):CURSOR0,21:PR
INTA(4):RETURN
5040 ON P(1)GOSUB 5042,5044,5046,5048:GOTO2900
5042 R1=6:GOSUB5200:R1=11:GOSUB5200:R1=16:GOSUB5050:R1=1:GOSUB5200:RETURN
5044 R1=11:GOSUB5200:R1=16:GOSUB5050:R1=1:GOSUB5200:R1=6:GOSUB5200:RETURN
5046 R1=16:GOSUB5050:R1=1:GOSUB5200:R1=6:GOSUB5200:R1=11:GOSUB5200:RETURN
5048 R1=1:GOSUB5200:R1=6:GOSUB5200:R1=11:GOSUB5200:R1=16:GOSUB5050:RETURN
5050 REM Player:CURSOR3,17:A$=STRING$(CHR$(3),8):PRINTA$(3,1):A$;"1":A$;"3":A$;"
4"1A$;"5"
5070 P=P+16TO 20:R=P-15
5080 CURSOR5,20:PRINTR;"ハナヲカキマシ (Y/N)?" :GET Z$
5090 IF Z$="N" THEN MUSIC"+B":NEXTP :GOTO5120
5100 IF Z$="Y" GOSUB2200:A(I,J)=21:E(R)=1 :Y=18:GOSUB2300:NEXTP:GOTO5120
5110 GOTO 5080
5120 CURSOR5,20:PRINT" " :MUSIC"+A+B+A+B"
5130 FOR R=1TO5:IFE(R)=1 THEN P=P+15:Y=18:GOSUB200:GOSUB2000
5140 NEXT R :FOR R=1 TO4:E(R)=0:NEXTR
5150 CURSOR3,17:A$=STRING$(CHR$(3),8):PRINTA$(3,1):A$;"-":A$;"-":A$;"-":A$;"-":A$;"-":R
ETURN
5200 REM ハナヲカキマシ:P=R1:GOSUB6000:IFB(0,S)>3THEN RETURN
5210 IFB(0,S)=2 THEN 5300
5220 IFB(0,S)=0 THEN 5400
5230 FOR P=R1 TOR1+4:GOSUB2200:IF A(0,J)<>1 THEN5250
5240 E(P-R1+1)=1
5250 NEXTP
5260 IF(RND(3)+RND(5))>1 THEN GOTO 5350
5270 FORJ=14TO2 STEP-1:FORI=1TO4:P=A(I,J):IFP >=R1THEN IFP <R1+5THEN IF A(0,J)=
1 THEN P1=A(I,J):GOTO5290
5280 NEXTI:NEXTJ:STOP
5290 E(P1-R1+1)=0:GOTO5350
5300 FOR P=R1 TOR1+4:GOSUB2200:IF A(0,J)<>1 THEN5320
5310 E(P-R1+1)=1:GOTO5350
5320 NEXTP
5330 PRINT
5340 PRINT" 174 / カキマシハ 1000 トムマツ"
5350 IF R1=1 THEN Y1=0:GOTO5355
5351 IF R1<>1 THEN Y1=6*INT(R1/5)
5355 P=R1-1:FOR R=1TO5:IFE(R)=1 THEN P=P+1:GOSUB2200:A(I,J)=21:Y=Y1:GOSUB2300
5360 NEXT R
5370 P=R1-1:FOR R=1TO5:IFE(R)=1 THEN P=P+1:GOSUB200 :Y=Y1:GOSUB2100
5380 NEXT R :FOR R=1 TO5:E(R)=0:NEXTR:RETURN
5400 GOSUB6370:IFC<4THEN 5470
5410 FOR J=14 TO 2 STEP-1:IF J<=A THEN IF J>=A-4 THEN 5430
5420 IF A(0,J)=1 THEN B1=J
5430 NEXT J
5440 FOR I=1 TO 4:IF A(I,B1)>=R1 THEN IF A(I,B1)<R1+5 THEN A=I:GOTO5460
5450 NEXTI
5460 E(A(A,B1)-R1+1)=1:GOTO 5350
5470 GOSUB 6430:IF C<4 THEN 5230
5480 FOR I=1 TO4 :IFA(I,15)=1 THEN A=I
5490 A(I,15)=0:NEXT I
5500 FOR J=2 TO14:IFA(A,J)>=R1 THEN IF A(A,J)<R1+5 THEN B1=J:GOTO5460
5510 NEXT J
6000 DIM B(4,4)
6020 S=1:IFP=6 THENS=2
6030 IF P=11 THEN S=3
6040 IF P=16 THEN S=4
6060 FOR I=0 TO 4:B(I,S)=0:NEXTI
6070 FOR J=14 TO2 STEP -1:A(0,J)=0
6080 FOR I=1 TO4
6090 IF A(I,J)>=P THEN IF A(I,J)<P+5 THEN A(0,J)=A(0,J)+1
6100 IF A(0,J)>B(0,S)THEN B(0,S)=A(0,J):B(1,S)=J:B(2,S)=I
6110 NEXT I:NEXTJ
6120 ON B(0,S)GOTO 6200,6170,6140,6130
6130 REM 4CARD: B(0,S)=7:RETURN
6140 REM 3CARD: J=14:GOSUB 6300
6150 IF B(3,S)<>0THENB(0,S)=6
6160 RETURN
6170 REM 1PARE: J=B(1,S)-1:GOSUB 6300
6180 IF B(3,S)=0 THEN B(0,S)=1
6190 RETURN
6200 REM Nothing: B(0,S)=0:GOSUB 6370
6210 IF C=5 THEN B(0,S)=4 :B(1,S)=A:B(3,S)=A-4
6220 GOSUB 6430
6230 IF C<>5 THEN RETURN
6240 B(0,S)=B(0,S)+5
6250 GOSUB 2200 :F=I :B(1,S)=5-I
6260 IF P>5 THEN IFP<11THEN B(3,S)=J :RETURN
6270 IF P>10THEN IFP<16THEN B(3,S)=J:RETURN
6280 IF P>15THEN IFP<21THEN B(3,S)=J:RETURN
6290 B(3,S)=J:RETURN
6300 REM Other 1PARE : IF J=1 THEN RETURN
6310 FOR J=J TO2 STEP-1
6320 IF A(0,J)=2 THEN B(3,S)=J:GOTO6340

```

```

6330 NEXT J:RETURN
6340 FOR I=1 TO 4
6350 IF A(I,J)>=P THEN IFA(I,J)<P+5 THEN B(4,S)=1:RETURN
6360 NEXT I:RETURN
6370 REM ステートC=0
6380 FOR R=2 TO 10:D=0
6390 FOR J=R TO R+4
6400 IF A(0,J)=1 THEN D=D+1
6410 IF D>C THEN C=D:A=J
6420 NEXTJ:NEXTR:RETURN
6430 REM フラッシュ:C=0:A(0,15)=0
6440 FOR I=1 TO 4:A(I,15)=0
6450 FOR J=2 TO 14
6460 IF A(I,J)>=P THEN IF A(I,J)<P+5 THEN A(I,15)=A(I,15)+1
6470 NEXTJ:IF A(I,15)>C THEN C=A(I,15)
6480 NEXTI:RETURN
6500 CURSOR68,Y:DN B(0,S)+1 GOTO 6510,6520,6530,6540,6550,6560,6570,6580,6590,
6590
6510 PRINT"Nothing":B(1,S)=0:B(3,S)=0:RETURN
6520 PRINT"1 ^?":B(1,S):B(3,S)=0:RETURN
6530 PRINT"2 ^?":B(1,S):B(3,S)=0:RETURN
6540 PRINT"3 カート":B(1,S):B(3,S)=0:RETURN
6550 PRINT"スタート":B(3,S)=1:~:B(1,S):RETURN
6560 PRINT"フラッシュ":FORI=1 TO4:IFF=TTHEND=T+132:PRINTCHR$(0):B(3,S)=0:RETURN
6585 NEXTI
6570 PRINT"7カレ?":B(1,S):B(3,S):RETURN
6580 PRINT"4 カート":B(1,S):RETURN
6590 PRINT"スタート フラッシュ":FOR T=1 TO4:IFF=TTHEND=T+132:CURSOR70,Y+1:PRINTCHR$(0)
)B(3,S)-4:"~":B(3,S):RETURN
6600 NEXT T
6700 CURSOR71,1:PRINT"スタート":MUSIC"5A5G5B5R5A5G5B":CURSOR71,1:PRINT"
"
:FORGO=1TO5:FORG1=1TO3
6710 IFC(G1)>=10THENP=(G1-1)*5+G0:GOSUB2200:X=(G0-1)*9+7:Y=(G1-1)*6:GOSUB2003
6720 NEXTG1:NEXTGO
7000 P=16:GOSUB6000:W1=0
7010 FOR W=1 TO 4:Y=(W-1)*6+1:IF C(W)>=10 THEN S=W:GOSUB 6500:NEXTW:GOTO7030
7020 W1=W+1:NEXTW
7030 IFW1=4THEN Y=21:CURSOR55,Y:PRINT"          カレ!!!!!!":MUSIC"5A5R5A":GOTO
B000
7100 FORV=1TO4:V1=0:V2=0:V3=0
7110 IFC(V)=5THEN NEXTV:GOTOB000
7120 FOR N=1TO4
7130 IF V=N THEN NEXTN:GOTO7180
7140 IF C(N)=5 THEN V3=V3+1:NEXTN:GOTO7180
7150 IF B(0,V)>B(0,N) THEN V1=V1+1
7160 IF B(0,V)=B(0,N) THEN V2=V2+1
7170 NEXTN
7180 IF V1>=3-V3 THEN 7230
7190 IF V1=2-V3 THEN IFV2=1 THEN 7300
7200 IF V1=1-V3 THEN IFV2=2 THEN 7300
7210 IF V2>=3-V3 THEN 7300
7220 NEXT V
7230 IF V=4 THEN 7250
7240 Y=(V-1)*6+1:CURSOR52,Y:PRINT"スタート ステート 1カレ":CONSOLE:N:MUSIC"-D+A-D+A-D+A
-D+A":CONSOLE:GOTO7260
7250 CURSOR 52,19:PRINT"スタート ステート 1カレ":MUSIC"+A+B+A+B+A+B+A+B"
7260 A(V)=A(V)+B(1)+B(2)+B(3)+B(4):A5=B(1)+B(2)+B(3)+B(4)-B(V):CURSOR53,(V-1)*6
+4:PRINT"スタート ステート 1カレ":FOR O=1TO4:A(O)=A(O)-B(O):NEXTO:GOSUB5020:GOTOB000
7300 DIMV(4):FORS=1 TO4:V(S)=0:IFC(S)>=10THENIFB(0,V)=B(0,S) THENV(S)=10
7310 NEXTS
7400 FORS=1TO4:S1=0:S2=0:S3=0
7410 IFV(S)=0THEN NEXTS
7420 FOR N=1TO4
7430 IF S=N THEN NEXTN:GOTO7480
7440 IF V(N)=0 THEN S3=S3+1:NEXTN:GOTO7480
7450 IF B(1,S)>B(1,N) THEN S1=S1+1
7460 IF B(1,S)=B(1,N) THEN S2=S2+1
7470 NEXTN
7480 IF S1>=3-S3 THEN V=S:GOTO7230
7490 IF S1=2-S3 THEN IFS2=1 THEN 7600
7500 IF S1=1-S3 THEN IFS2=2 THEN 7600
7510 IF S2>=3-S3THEN 7600
7520 NEXT S
7530 IF V=4 THEN 7250
7600 DIMS(4):FORV=1 TO4:S(V)=0:IFV(V)=10THEN IFB(1,S)=B(1,V) THENS(V)=10
7610 NEXTV
7700 FORV=1TO4:V1=0:V2=0:V3=0
7710 IFS(V)=0THEN NEXTV
7720 FOR N=1TO4
7730 IF V=N THEN NEXTN:GOTO7780
7740 IF S(N)=0 THEN V3=V3+1:NEXTN:GOTO7780
7750 IF B(3,V)>B(3,N) THEN V1=V1+1
7760 IF B(3,V)=B(3,N) THEN V2=V2+1
7770 NEXTN
7780 IF V1>=3-V3 THEN 7230
7790 IF V1=2-V3 THEN IFV2=1 THEN 7900

```

```

7800 IF V1=1-V3 THEN IFV2=2 THEN 7900
7810 IF V2=3-V3 THEN 7900
7820 NEXT V
7900 DIM V(4):FOR S=1 TO 4:V(S)=0:IFS(S)=10 THEN IFB(3,V)=B(3,S) THEN V(S)=10
7910 NEXTS:A3=0
7915 FOR U=1 TO 4:IF V(U)=10 THEN IFB(0,U)=0 THEN Y=21:CURSOR 6,B,Y:PRINT"フカレ!!!!!!":H
USIC"SA5R5A":GOTO 8000
7916 NEXTU
7920 FOR S=1 TO 4:IF V(S)=10 THEN Y=(S-1)*6+1:CURSOR 52,Y:PRINT"トワツ イ 1...":NEX
T S:GOTO 7940
7930 A(S)=A(S)-B(S):A3=A3+B(S):NEXTS
7940 FOR S=1 TO 4:IF V(S)=10 THEN Y=(S-1)*6+1:CURSOR 53,Y+3:PRINT"ハイト ハ ";INT(
A3/V2)"トハ":A(S)=A(S)+INT(A3/V2)
7950 NEXTS:GOSUB 5020
7960 MUSIC"+G+G+G+G+G+G+G+G+G+G+"
8000 CURSOR 53,23:PRINT" [ Hit any key ] "
8005 GETN$
8010 IF N$="" THEN 8000
8012 FOR I=1 TO 4:IF A(I)>25500 THEN CONSOLEC40,N:V=I:GOSUB 10380:CURSOR 11,7:PRI
N
8014 NEXT I
8020 LIMIT$F000
8022 DIM J(4):FOR I=1 TO 4:IF A(I)<0 THEN J(I)=1:A(I)=-A(I):NEXT I:GOTO 8026
8024 J(I)=0:NEXT I
8026 POKE$F110,J(1):POKE$F120,J(2):POKE$F130,J(3):POKE$F140,J(4)
8030 POKE$F050,K1:POKE$F010,INT(A(1)/100):POKE$F020,INT(A(2)/100):POKE$F030,INT
(A(3)/100):POKE$F040,INT(A(4)/100):POKE$F060,K0
8040 CLR
8050 DIM A(4):K1=PEEK($F050):K0=PEEK($F060):A(1)=PEEK($F010)*100:A(2)=PEEK($F02
0)*100:A(3)=PEEK($F030)*100:A(4)=PEEK($F040)*100:K1=K1+1
8052 DIM J(4):J(1)=PEEK($F110):J(2)=PEEK($F120):J(3)=PEEK($F130):J(4)=PEEK($F14
0)
8054 FOR I=1 TO 4:IF J(I)=1 THEN A(I)=-A(I)
8056 NEXT I
8058 IF K1=K0+1 THEN 10000
8060 GOTO 15
9000 REM 15:CONSOLEC40,N:CURSOR 0,5:GOSUB 9010:L=1
9005 CURSOR 0,5:ONL GOSUB 9020,9040,9060,9080:L=L+1:IF L=5 THEN 9110
9006 GOTO 9005
9010 PRINTSTRING$(CHR$(147),6):FOR I=1 TO 4:PRINTCHR$(147)+STRING$(CHR$(3),5)+CHR$
(147):NEXT I:PRINTSTRING$(CHR$(147),6):FOR I=1 TO 5:PRINTCHR$(147):NEXT I:RETURN
9020 PRINTSTRING$(CHR$(3),9)+STRING$(CHR$(147),5):FOR I=1 TO 9:PRINTSTRING$(CHR$(3
),8)+CHR$(147)+STRING$(CHR$(3),5)+CHR$(147):NEXT I
9030 PRINTSTRING$(CHR$(3),9)+STRING$(CHR$(147),5):RETURN
9040 FOR I=1 TO 6:PRINTSTRING$(CHR$(3),16)+CHR$(147)+STRING$(CHR$(3),6-I)+CHR$(14
7):NEXT I
9050 FOR I=1 TO 5:PRINTSTRING$(CHR$(3),16)+CHR$(147)+STRING$(CHR$(3),I)+CHR$(147
):NEXT I:RETURN
9060 FOR I=1 TO 2:PRINTSTRING$(CHR$(3),24)+STRING$(CHR$(147),7):FOR I=1 TO 4:PRINTSTR
ING$(CHR$(3),24)+CHR$(147):NEXT I:NEXTU
9070 PRINTSTRING$(CHR$(3),24)+STRING$(CHR$(147),7):RETURN
9080 PRINTSTRING$(CHR$(3),32)+STRING$(CHR$(147),6):FOR I=1 TO 4:PRINTSTRING$(CHR$(
3),32)+CHR$(147)+STRING$(CHR$(3),5)+CHR$(147):NEXT I
9090 PRINTSTRING$(CHR$(3),32)+STRING$(CHR$(147),6)
9100 FOR I=1 TO 5:PRINTSTRING$(CHR$(3),32)+CHR$(147)+STRING$(CHR$(3),I)+CHR$(147
):NEXT I:RETURN
9110 CURSOR 2,18:PRINT"フカハ コンピューター ミニ ミカエカ !!!!!???"
9120 CURSOR 5,21:PRINT"モロモロ カークフェイス ミカエカ !!" :MUSIC"+A+A+A"
9130 CURSOR 10,24:PRINT" [Hit any key]"
9140 GETN$
9150 IF N$="" THEN 9140
9200 REM 15:CONSOLEC40,N:CURSOR 0,2
9210 PRINT" フノ ヲヨノ シュン ハ ツキ ノ トワリ"
9220 PRINT"
"
9230 PRINT" 1. ストレート フラッシュ (♠>♥>♣>♠)"
9240 PRINT" 2. フェーカート"
9250 PRINT" 3. フラッシュ (♠>♥>♣>♠)"
9260 PRINT" 4. ストレート"
9270 PRINT" 5. フェーカート"
9280 PRINT" 6. スリーカード"
9290 PRINT" 7. リーネアチ"
9300 PRINT" 8. ワン ハート"
9310 PRINT"
"
9320 PRINT" モロモロ ハ 5000 トハ"
9330 PRINT" 1カ / カケン ハ 1000 トハ マテ"
9340 PRINT"
"
9350 PRINT" フカハ ショップ ニ シマスカ (1 ~ 10)":INPUTN
9360 IF N=0 THEN 9350
9370 K0=N
9380 PRINT"
"
9390 PRINT"
"
9400 PRINT"
"
9410 PRINT" ソレハ カン ハ マテ !!!!!!!"
9420 PRINT"
"
9430 PRINT"
"
9440 PRINT"

```

```

9450 PRINT"                [Hit any key] "
9460 GET N$:
9470 IF N$="" THEN 9460
9480 GOTO 12
10000 CONSOLEC40,N: DIM B(4): FOR I=1 TO 4: B(I)=10: NEXT I
10010 CURSOR 8,2: PRINT"   ヒットキー 11111": CURSOR 10,3: PRINT K0;" カイ タイセン セイロキ" : M
USIC" +GF+GF+A+B+A+B+-5C+5B"
10100 FORM=1 TO 3: FOR V=1 TO 4: V1=0: V2=0: V3=0
10110 IF B(V)=<M-1 THEN 10220
10120 FOR N=1 TO 4
10130 IF V=N THEN 10170
10140 IF B(N)=<M-1 THEN V3=V3+1: GOTO 10170
10150 IF A(V)>A(N) THEN V1=V1+1
10160 IF A(V)=A(N) THEN V2=V2+1
10170 NEXT N
10180 IF V1>3-V3 THEN B(V)=M
10190 IF V1=2-V3 THEN IF V2=1 THEN B(V)=M
10200 IF V1=1-V3 THEN IF V2=2 THEN B(V)=M
10210 IF V2=3 THEN Y=10: CURSOR 12,Y: PRINT" トウツ ショイ" : GOTO 10500
10220 NEXT V: NEXT M: F=0: G=0: H=0: CURSOR 0,7
10300 FOR V=1 TO 4: IF B(V)=1 GOSUB 10400: NEXT V: GOTO 10320
10310 F=F+1: NEXT V
10320 FOR V=1 TO 4: IF B(V)=2 GOSUB 10410: NEXT V: GOTO 10340
10330 IF B(V)>2 THEN G=1: NEXT V: GOTO 10340
10335 NEXT V
10340 FOR V=1 TO 4: IF B(V)=3 GOSUB 10420: NEXT V: GOTO 10360
10350 IF B(V)=10 THEN H=1: NEXT V: GOTO 10360
10355 NEXT V
10360 FOR V=1 TO 4: IF B(V)=10 GOSUB 10430
10370 NEXT V: GOTO 10500
10380 ON V GOTO 10382, 10384, 10386, 10388
10382 E$="M-Z": RETURN
10384 E$="M-C": RETURN
10386 E$="M-B": RETURN
10388 E$="A-F": RETURN
10400 GOSUB 10380: PRINT STRING$(CHR$(3), 13); "イ" : E$: " " : A(V); "ト" : MUSIC" +A+
B": PRINT: RETURN
10410 GOSUB 10380: PRINT STRING$(CHR$(3), 12); 5-F; "イ" : E$: " " : A(V); "ト" : MUSIC"
+C+D": PRINT: RETURN
10420 GOSUB 10380: PRINT STRING$(CHR$(3), 12); 5-G; "イ" : E$: " " : A(V); "ト" : MUSIC"
AB": PRINT: RETURN
10430 GOSUB 10380: PRINT STRING$(CHR$(3), 12); 5-H; "イ" : E$: " " : A(V); "ト" : MUSIC"
-C-D": PRINT: RETURN
10500 CURSOR 10,21: PRINT" TRY AGAIN? (Y/N)"
10510 GET N$
10520 IF N$="Y" THEN CL: GOTO 2
10530 IF N$="N" THEN CONSOLEC40,N: END
10540 GOTO 10510
10550 END

```

MZ-80B

チェス・プログラム

CHES PROGRAM

RD-Q7

このプログラムは、次の3つの目的をもって作りました。

①CHESS

②ENDINGの練習

③チェス問題を解く

①②③の3つでスタートさせると、③の3つのゲームを選択するスクリーンにやってきました。写真1。

TOMUC ● 馬場隆信 / COMPAC

1

CHESS

〔C〕キーを押すと、コンピュータとチェスの対戦をすることになりますが、その前に先手が後手かを選びます (写真2)。

次にLEVELを選びます (写真3)。僕としては〔3〕を中心にプログラムを作ったつもりですが、最初のうちはレベル〔1〕で序盤定跡を覚えてから、〔3〕に進むのがいいと思います。

チェスの駒の動かし方は、国際表記法を使っています。チェスの表記法には大きく分けて2つあり、他の1つは英米書式です。私達がチェスの本を読むときは、たいいてい英語ですから、英米書式も覚えておく必要があるのですが、最近ではアメリカでも国際表記法を使い始めているので、このプログラムでも国際表記法を使いました。

図2を見てください。g 2のビショップを、f 3へ動かすときは、Bf3と書きます。キー入力では、〔B〕〔f〕〔3〕となります (図3)。

e 1のナイトを、f 3へ動かすときは、Nf3と書きます。キー入力は〔N〕〔f〕〔3〕です。では、e 1のナイトをc 2へ動かすときはどうでしょうか。このときはNec2 (〔N〕〔e〕〔c〕〔2〕) となります。なぜかと言うと、a 1のナイトもc 2へ進めるために、どちらのナイトかを区別する必要があります。この場合、R 1 d 3あるいは、Rdd3と書きます。キー入力は〔R〕〔1〕〔d〕〔3〕です。

では、d 1にあるルークをd 3へ進める場合はどうでしょうか。a 3にあるルークもd 3へ進むことができるために、どちらのルークかを区別する必要があります。この場合は、R 1 d 3あるいは、Rdd3と書きます。キー入力は〔R〕〔1〕〔d〕〔3〕です。

さて、ボーンの動かし方は少し変わっています。たとえば、g 3のボーンをg 4に動かすときには、Pg4とせず、単にg 4とします。また、a 4のボーンでb 5の敵駒を取る場合は、ab5と書きます。これは、a ファイルのボーンで、b 5の駒を取るということを表わしています。

キャスリングは、K側をO-O、Q側をO-O-Oと書くのですが、キー入力では、〔k〕をキング・サイド・キャスリング、〔q〕をクイン・サイド・キャスリングとしました。

写真1



写真2



写真3



写真4



なお、キー入力の後には、必ず **[CR]** キーを押してください。

プリンタへの出力

ゲーム進行中に、次の命令が使えます。

[T M CR] : 画面上1行にメッセージを書き、**[CR]** でプリンタへ出力。

[T 1 CR] : 手筋を国際表記法で印字する。

2 ENDING練習

写真1で **[E]** のキーを押すと写真4になり、どんなENDINGを希望するかを **[1] ~ [9]** のキーで指定します。すると、乱数で適当に駒を並べて問題を作ってくれます。後は1のCHESSと同じです。

相手のキングを詰ますには、最低限、図4のような駒が必要で、これを、メイティング・マテリアルといいます。このブロックは、乱数の作り方の失敗で、同じパターンの問題が出てしまいます。

[X CR] キーで駒設定ブロックを呼んで問題を作り、**[X CR]** で戻してENDING練習してください。

3 チェス問題を解く

[P] キーを押すと、このブロックにきます(写真5)。まず、チェス・ボードに駒を並べます。これはメイン・キーボードを使ってください。**[P f 3]** と **[CR]** でポーンの駒がf3のマス目に置かれます。

次に、深度を設定します。**[5]** キーを押すと、DEPTH?と聞いてくるので、2手詰のときは **[4]** を、3手詰のときは **[6]** を押してください(写真6)。

そして、**[V]** キーを押すと、答を画面右側に表示して、駒も1回分を動かしてくれます。

次に、**[A]** キーを押すと白黒を入れ換えてくれます。そして、再び **[V]** キーを押します。これを繰り返すと、解答の全手筋が解ることになります。これは、詰めチェスのみではなく、NEXTムーブでも使えます。

[I] で写真1へ戻ります。

4 序盤データ・エディタについて

1のCHESSから、**[O CR]** でこのブロックにきます。これにより、序盤データを自分で作ることができ、練習したい手筋を作れるわけです(写真7)。

このブロックは、チェス盤の換わりもしてくれるので、手筋の研究にとっても便利です。

図2

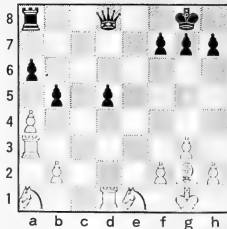
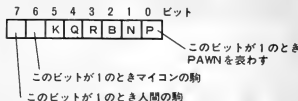


図3 駒コード



このビットが1のとき
PAWNを表す

このビットが1のときマイコンの駒

このビットが1のとき人間の駒

-	0	1	2	3	4
5	6	7	8	DEL	
CR	A	B	C	K (0-0)	Q (0-0-0)
	D	E	F	G	H

写真5

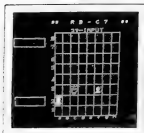


写真6

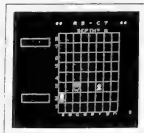


写真7

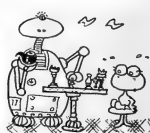
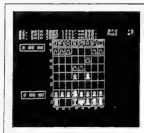


図4 メイティング・マテリアル

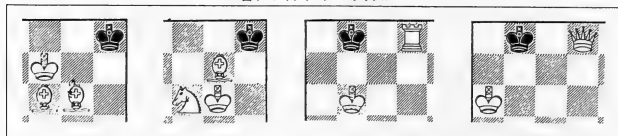
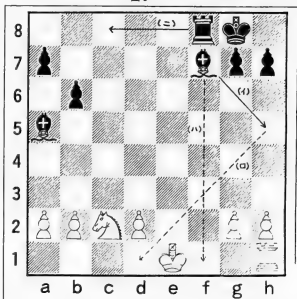


図9



す、次に自のキングを動かしてみます。すると、点線の通っているマス目でチェックを受けることになります。この場合4回です。この回数に比例して、この評価点は高くなるわけです。

ATTACK：攻撃

図9において、ルークを矢印(二)のように動かすことは、C2のナイトを攻撃したことになります。当然、逃げられて駒を取ることはできないのですが、それなりの有効点を与えるわけです。

STRUCTURE：構成

a7のポーンがb6のポーンを守っています(図9)。また、b6のポーンとa5のビショップとは互いに守り合っています。こういう手が積み重なって、堅固な構成ができあがるわけです。

POSITION：駒位置

(一)絶対駒位置

たとえば、将棋のプログラムなどで、穴熊を使うとか、矢倉を使うとかのときに、その動きに評価点を付けます。

(二)相対駒位置

これは非常に面白い評価点なのですが、処理時間を多く取りそうです。たとえば、ルークを二重に使うダブル・ルーク(将棋でいう二枚飛車)、同基などで、大桂馬に何点、小桂馬に何点というように付けたり、簡単な詰め碁の型やコンビネーションに評価点を与えます。

一 キーを押すと、1手分、駒の動きを戻してくれます。
 二 キーは、1手戻すと同時に、序盤データのポイントも1手分戻し、訂正に使います。チェスの1手は将棋の2手にあたるので、これらのキーを使うときは、白黒の1手を動かした後に押す必要があります。

三 キーは、ポイントの変更に使います。データを継ぎ足すのに使います。

なお、コマンドの実行には、[CR]を押すことが必要です。いろいろと図1を見ながら試してみてください。

5 チェスのプログラム

チェスのプログラムを作る場合、まず、チェス・ボードを用意します。これは適当なRAM上に決めるのですが、今回はA 0mn番地(m, n = 0 ~ 7)の64バイトを使っています。次に、駒コードも適当に決めて、チェス盤のRAM上に並べます。そして、実際に駒をあれこれ動かしてみ、次の1手を決めてくるわけです。

たとえば、1.5手読みの場合、まず、自分の駒を動かし、次に相手の駒を動かし、そして、もう1回、自分の駒を動かして初めて、その1.5手読みの評価点が決まります。そして、この評価点の最大のを、次の1手として駒を動かしてくるのです。

6 評価点の種類

では、どのようにして評価点を付けていくのかを考えてみたいと思います。これらの判断基準を多く使えば、それだけ良いプログラムができるわけですが、逆に処理速度は遅くなってしまいます。また、先を何手も深く読めば読むほど、プログラムは強くなり、それにより不必要になってくる判断基準も出てきます。

GAIN：駒の獲得

相手の駒を取ったときに与える評価点で、最優先に近い点を与えます。一般に、駒の価値は、P : N : B : R : Q = 1 : 3 : 3 : 5 : 9 というのがよく知られていますが、このプログラムでは1 : 3 : 3 : 7 : 15を使っています。これは、実際に走らせてみて調整した比率です。

CONCENTRATION：駒の集結

たとえば、図9の場合、f7のビショップを矢印(一)のように動かしたとき、ビショップのきき筋が点線(一)のように通り、キングは動きにくくなります。また、ルークのきき筋も点線(二)のように通り、一層キングは動きにくくなります。このような手に与える評価点がこれです。

この判定はプログラムの中で次のようにして見つけます。たとえば、ビショップの場合、まず(一)のように動かしま

図 9

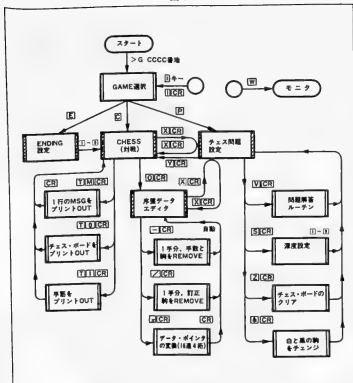
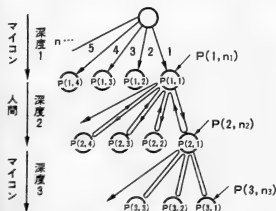


図10

TREE of minimax

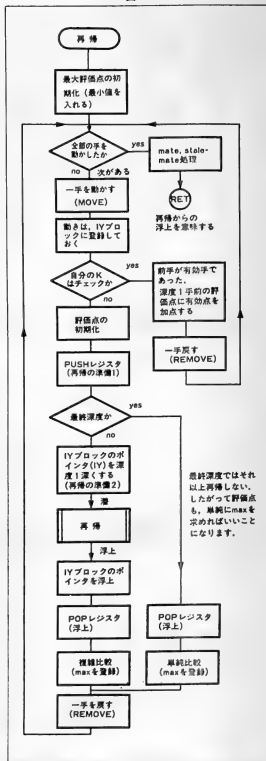
(1.5手読みの場合、深度3まで、 $P(m, n)$ の m は深度)

$P(1, \max)$ は、真の $P(1, n_1)$ の中で最大の値を取る手を意味する。
 真の $P(1, n_1) = P(1, n_1) - |P(2, \max) - P(3, \max)|$
 $= P(1, n_1) - P(2, \max) + P(3, \max)$

その他

エースリングなどの特殊な手を与えます。他にもいろいろ考えられると思いますが、いま式での評価点の付け方が単純になると思います。他にも、駒の動きやすさを考慮したMOVINGやDEFENCE関係がありそうです。

図 8



7 Tree of mini-max

mini-maxに関する正式な知識はないのですが、以下に述べるのがそれだと思います(図8、図10、図11)。

まず、マイコンで n 手読みのチェス・プログラムを作ろうとする場合、メモリ容量の制限で、同じようなルーチンをいくつも作るのを省くために、再帰型のmini-maxルーチ

図11

1 Yブロック(16バイト単位のワーク・エリア)

1 Y + 0 0	L (0) 動かす駒の 下位番地
0 1	BOX (0) その駒
1 Y + 0 2	L (N) 駒の目的 地の番地
0 3	BOX (N) そのマス目 にある駒
1 Y + 0 4	対応する深度
0 5	フラグ
0 6	アンバサン番地
	その他
1 Y + 0 A	評価点
0 B	"
1 Y + 0 C	その深度における 最大評価点
0 D	
1 Y + 0 E	L (0) max
0 F	L (N) max

マスの
印番地

図12

1 Yブロック用エリア(A100~16バイト単位)

※ 1 Yレジスタでポインタする。深度(再帰回数)と呼応して動く。

A 100	A 110	A 120	A 130	A 140
深度 0 用 (人間用用の ワークエリア)	深度 1 (マイコン用 minimax 用エリア)	深度 2	深度 3	深度 4

図 6

図 5

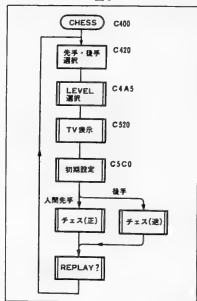
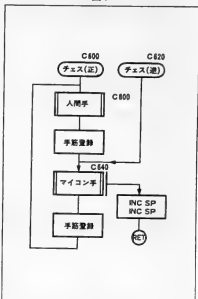


図 7



ンを作る必要があります(図8)。

1.5手読みを前提とした場合、図10のように(マイコン)(人間)(マイコン)と、3回、駒を動かしてみることになります(チェスの1手は将棋の2手)。

まず、1回、駒を動かすわけですが、この動きには次の3つの要素が含まれています。

- ① どの駒を動かすか?
- ② どの方向に動かすか?
- ③ その方向に何ステップ動かすか?

さて、最初の動きP(1,1)によって得られる評価点は真の評価点ではありません。なぜなら、深度3まで読むのが1.5手読みだからです。そのため、とりあえずこの評価点をワーク・エリアに登録しておく必要があります。これが、図11の1 Yブロックです。この1 Yブロックは、深度1用、深度2用……とn手分必要です(図12)。そして、再帰するたびに、その深度に合わせたブロックを使います。

結局、真の評価点P(1,n)は図10の下式で求まることになります。

8 α - β Algorithm (枝切り)

枝切りに関しては、私の力量不足から、このプログラムの中では具体化できなかったというのが事実です。ただ1つ使った枝切りは、1手動かしたときに自分のキングがチェックになったときです。残さず手や、相手のきき筋に自

分のキングを動かしたときがこれです。図9で、ポーンをd3へ動かしたときです。もし、このように動かしたら、当然相手はキングを取って勝ってしまうので、この手は禁じ手となり、また、この手以後の深度の手は考慮に入れる必要なくなるわけで、これが枝切りの一つとなります。

(注) 人間側の入力に関して、禁止手のチェックを行っていません(キングが一度動いた後でもキャスリングできる)、打ち手の側で判断してください。

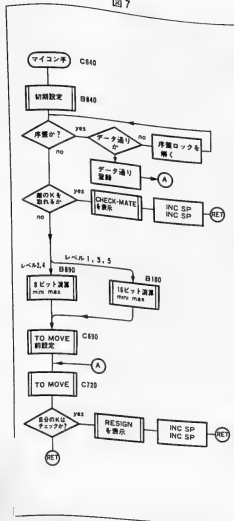
9 Computer Chess

歳末ではコンピュータのチェス大会が盛んなようです。実態は知りませんが、BYTEなどに載っているんじゃないでしょうか。見たことは、ないので解りませんが。

以前、Sargon 2.5というチェス・プログラムについて少し話したことがあります。Sargon 3というのができています。これはおそらく3手先を読むのだと思うのですが、問題なのは、3手先をどの程度の時間で読むのかということと、どんなCPUで何MHzで走っているのかということです。

いま、僕が残念に思っているのは、その程度のプログラムを作ったのはいいのですが、どの程度の能力を持ったプログラムなのか解らないということなのです。このプログラムがどの程度の能力を知る方法はたまた1つです。つまり、他のチェス・プログラムと戦わせてみるということです。

図7



TIやAPPLEにはチェスのソフトがあるらしく、ぜひ戦わせてみたいものです。

さて、もう一度話を戻します。

Bellという人がMachine Plays Chessという本を書いています。この人の作ったチェス・プログラムは、1980年の世界コンピュータ・チェス選手権で優勝したそうです。

Bellというと、あなたも何かを連想するでしょう。そうです、Bell研究所を連想してしまいます。しかし、名前が同じだからといって現実検討抜きで結論付けるのは、早合点というものですから、僕は結論を延期していました。しかし、どうやらそのようです（確信はありません）。Bellというプログラム・ネームで、マシンはかなり高速で走る大型コンピュータのようです。

10

あとがき

チェスのプログラムを作って、一番感じるのは処理速度で、8ビットのCPUでは限界を感じてしまいます。早く、8MHzで走る68000でプログラムを組んでみたいものです。このとき、普通の人とはまったく勝てなくなると思います。

人間同志のチェス大会はどうでしょうか？ FIDE(世界チェス連盟)が2年に一度、チェス・オリンピックを開い

図13

レジスタ名	使用目的
Bレジスタ	深度
DEレジスタ	進行方向値のポインタ
HLレジスタ	チェス盤のマス目のポインタ
B'Cレジスタ	IYレジスタ加算用
D'Eレジスタ	" 減算用
IYレジスタ	IYブロックのポインタ
IXレジスタ	手筋登録用など

図15

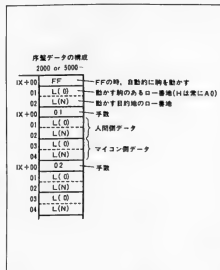


図18

評価点の例 (16ビット演算を使う場合)

No.	評価点	GAIN	DEFENCE	STRUCTURE	ATTACK	CONCENTRATION
PAWN	00A0	0000	PAWNのみ P×0001	0000	0000	
NIGHT	0200	0000		0000	0000	
BISHOP	0202	0001		0000	0000	
ROOK	0409	0003		000C	000E	マス目1つにつき
QUEEN	0900	0005		0018	001E	001E
KING	1400+α	000A		001E+β	001E+β	(キングのみ)

注: α や β は深度によって値が変わる。たとえば、KINGは先に取った方が勝たぬから、深度が浅い程評価値は高くなる。

ています。日本には将棋や囲碁があるのでほとんど知られていませんが、かなり大きな国際大会です。1980年は地中海のマルタで開かれ、82箇国が参加しました。1982年はスイスのルツェルンで開かれる予定で、100箇国が参加するのではないかとされています。

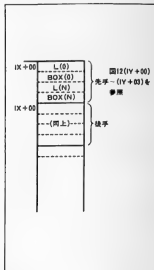
日本でチェスを楽しんでいるのは500人はもしかいません。そのうち、研究肌の人は100人程です。つまり、その気になれば、あなたも日本代表として世界大会への参加が可能なのです。特に、19才以下の世界ジュニア選手権大会ではそうです。

日本代表は毎回10名程で、日本チェス協会の選択大会で決められます。脳細胞に自信のある人は、試してみればどうでしょうか。

図14

B09E	キング番地(マイコン側)
B09F	" (人間側)
B0A0	フラグ: 人間先手……00
B0A1	LEVEL
B0A2	DEPTH (深度)
B0A3, 4	使用時間(マイコン側)
B0A6, 7	" (人間側)
B0A8	フラグ: 序盤時……00
B0A9, A	ポイント番地(序盤データ)
B0AB	現在の手筋
B0AC, D	ポイント番地(手筋ストア・エリア)
B0AE	手筋(プリンタ用)

図16



チェス・プログラム・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
2900	45	36	22	34	10	55	34	10	55	34	10	55	34	10	55	34	10
2910	35	22	43	10	31	53	43	10	31	53	43	10	31	53	43	10	31
2920	25	44	13	FA	04	13	FA	04	13	FA	04	13	FA	04	13	FA	04
2930	10	16	75	03	07	05	11	77	75	03	07	05	11	77	75	03	07
2940	09	FA	FA	34	26	10	75	05	23	07	75	05	23	07	75	05	23
2950	60	50	16	26	12	25	34	04	34	08	55	35	22	41	09	53	24
2960	34	22	34	12	25	34	04	34	08	55	35	22	41	09	53	24	1E0
2970	25	44	10	52	44	13	33	07	52	33	07	52	33	07	52	33	07
2980	25	09	74	07	04	15	10	72	63	13	33	11	75	53	22	41	169
2990	07	75	31	25	08	31	12	22	09	74	07	04	15	10	72	63	13
29A0	77	74	05	37	11	45	23	41	12	25	34	04	34	08	55	35	22
29B0	22	12	14	44	34	41	52	14	72	63	07	04	12	72	63	05	15
29C0	13	23	13	64	53	41	52	14	63	52	37	46	11	72	63	34	1F3
29D0	45	12	44	34	02	04	13	63	45	23	41	09	45	34	23	34	1C0
29E0	10	55	34	04	34	06	35	FA	08	55	35	22	41	09	53	24	19A
29F0	31	53	16	26	11	74	47	05	12	66	55	04	15	08	45	15	157
Sum	05	3E	AD	AB	22	66	05	B5	4F	B5	BB	50	0B	41	6C	0C	1F7

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum	
2E00	45	34	04	34	12	54	45	34	24	13	FA	04	05	32	12	74	17B	
2E10	65	34	54	13	65	54	33	43	10	74	74	34	44	11	53	44	149	
2E20	25	44	12	52	44	26	43	13	66	44	04	44	12	45	35	44	150	
2E30	52	13	64	52	26	15	11	45	35	44	53	12	62	53	26	15	17A	
2E40	11	53	44	12	66	44	12	66	44	12	66	44	12	66	44	23	15B	
2E50	42	34	44	33	43	09	45	34	22	34	10	74	56	17	37	11	46	145
2E60	37	26	37	09	72	63	06	25	10	FA	23	33	09	74	65	10	19	
2E70	06	25	10	46	26	25	06	08	76	55	06	09	75	66	02	16	158	
2E80	04	10	74	56	34	44	11	53	44	23	33	10	74	55	34	44	185	
2E90	11	53	44	23	33	11	45	35	44	53	47	26	53	47	26	53	148	
2EA0	46	36	34	44	11	53	44	25	44	10	45	35	26	15	11	53	130	
2EB0	36	15	06	10	72	63	34	44	11	53	44	25	44	09	72	63	190	
2EC0	34	45	10	63	45	23	33	11	46	36	33	43	10	52	64	22	172	
2ED0	34	09	45	34	02	04	10	74	56	23	34	09	46	36	34	44	146	
2EE0	10	55	47	26	17	10	53	44	25	44	08	76	64	23	33	08	159	
2EF0	45	26	35	15	09	76	55	06	25	10	75	66	02	04	10	55	10A	
Sum	01	16	40	41	47	7A	3C	34	72	3B	FF	4E	14	3B	D2	A0	17E	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
2A00	34	22	34	09	72	36	FA	10	FA	07	05	11	72	73	135		
2A10	25	46	09	72	63	FA	10	FA	07	37	06	45	35	02	1FE		
2A20	11	07	72	36	FA	08	54	13	09	74	63	33	25	13	14E		
2A30	10	FA	04	15	11	74	44	05	23	12	63	51	22	30	13	139	
2A40	44	46	30	51	13	55	34	04	12	75	22	20	11	55	141		
2A50	34	22	34	12	74	34	02	04	11	75	42	17	27	10	73	106	
2A60	17	27	11	36	47	32	12	47	65	32	65	13	74	65	34	17B	
2A70	44	10	75	32	12	74	04	34	12	55	34	17	27	13	10C		
2A80	11	55	43	04	11	63	55	05	41	09	FA	33	44	10	18A		
2A90	43	44	05	41	11	36	25	44	12	63	41	22	41	13	74	1A0	
2AA0	41	44	23	12	74	44	22	03	06	45	34	22	34	07	74	56	170
2AB0	34	55	08	74	25	35	45	34	12	35	04	33	25	13	16B		
2AC0	25	11	71	63	05	27	12	56	05	02	46	66	59	02	11	121	
2AD0	06	71	63	02	11	07	47	42	34	08	34	24	25	33	09	1F3	
2AE0	72	45	33	45	10	74	47	05	14	11	47	45	FA	06	75	11F	
2AF0	31	02	11	07	31	22	11	22	08	45	34	25	44	09	71	63	178
Sum	04	03	1E	3A	3B	B1	D2	E3	4B	83	CB	80	89	FA	7E	1B7	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
2F00	36	15	06	10	72	63	06	25	08	76	65	34	44	09	53	44	147
2F10	25	44	08	72	63	06	25	08	76	65	34	44	09	53	44	23	151
2F20	34	08	72	63	06	25	08	76	65	34	44	09	53	44	26	15	158
2F30	09	75	66	22	43	07	72	63	06	25	08	76	65	34	44	09	1CE
2F40	76	55	23	33	06	76	55	04	13	07	72	63	34	44	08	63	191
2F50	45	17	27	09	67	57	02	04	08	FA	23	33	09	74	65	03	141
2F60	44	10	75	32	12	74	04	34	12	55	34	17	27	13	10C		
2F70	25	08	72	63	06	25	08	76	65	34	44	09	53	44	26	15	152
2F80	06	25	08	72	63	06	25	08	76	65	34	44	09	53	44	23	106
2F90	76	64	04	13	07	67	57	17	37	06	45	34	23	34	07	72	153
2FA0	63	04	13	08	FA	06	25	07	76	55	06	23	34	08	67	57	146
2FB0	13	08	09	46	35	20	59	76	64	04	13	08	72	63	04	10	152
2FC0	13	08	72	63	05	23	09	FA	22	43	05	72	36	02	55	152	
2FD0	06	FA	FA	FA	07	66	56	04	13	05	72	63	FA	06	75	15C	
2FE0	FA	06	25	07	76	64	16	26	07	66	56	04	13	05	67	182	
2FF0	45	26	06	46	35	24	07	76	64	FA	04	75	53	12			
Sum	05	59	C3	D1	AC	3B	AF	25	6A	3F	45	B6	F7	BD	DA	1F	19E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
2B00	44	32	06	75	53	02	11	07	72	63	FA	03	65	55	01	1E5	
2B10	22	04	44	35	17	37	05	72	36	13	23	06	66	46	05	14	19B
2B20	07	74	63	37	06	38	26	15	25	09	55	46	34	08	47	07	11A
2B30	71	52	37	07	75	16	12	74	04	34	12	55	34	17	27	13	10C
2B40	56	22	43	08	71	52	37	06	47	67	47	37	46	08	36	25	19B
2B50	14	25	09	55	46	16	26	10	75	66	26	33	11	46	36	25	111
2B60	16	04	76	64	13	33	05	44	33	25	33	06	71	52	02	24	1FD
2B70	07	55	45	34	05	08	64	45	07	65	45	FA	07	13	1E		
2B80	72	36	05	14	04	04	FA	08	52	33	24	33	02	52	1F8		
2B90	35	22	33	07	72	63	FA	08	52	33	24	33	04	72	54	10B	
2BA0	13	33	05	44	33	25	33	06	44	36	05	14	06	54	63	02	1B2
2BB0	24	05	34	33	03	06	44	35	02	36	07	71	63	FA	1AE		
2BC0	03	72	63	45	35	44	33	05	44	33	02	65	55	11	72	73	11E
2BD0	75	42	11	31	07	42	53	10	10	20	68	63	54	01	13	09	712
2BE0	63	04	26	10	53	75	02	11	11	63	51	32	54	11	60	40	174
2BF0	FB	FB	10	54	32	13	32	11	74	56	26	15	09	60	40	31	1C1
Sum	17	E1	26	7B	75	66	EA	47	F9	5F	EB	15	55	52	4F	BF	1DF

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
3000	06	25	05	65	55	17	37	06	61	51	23	35	07	76	57	22	133
3010	41	05	53	44	25	17	37	06	74	44	04	15	05	61	51	22	137
3020	06	43	23	33	05	65	57	41	04	15	05	61	51	22	137	04	110
3030	66	46	17	37	05	46	36	02	35	06	63	53	FA	05	65	65	1CC
3040	55	37	46	06	55	16	26	04	67	52	02	35	05	65	66	46	159
3050	35	24	06	75	66	FA	04	61	51	23	35	07	76	57	22	61	1ED
3060	FA	06	65	45	34	44	07	FA	06	28	52	73	05	65	66	13	08
3070	66	46	17	37	05	46	36	02	35	06	63	53	FA	05	65	65	1CC
3080	64	56	35	26	09	67	47	02	04	09	75	31	10	20	10	12	12A
3090	22	15	22	09	76	55	02	04	10	75	64	25	33	10	67	47	130
30A0	17	27	07	76	64	23	03	08	64	36	35	13	09	65	05	05	147
30B0	32	10	75	31	10	20	07	65	45	05	08	14	04	22	02	07	193
30C0	76	55	34	44	08	55	27	13	03	04	75	31	02	35	05	63	178
30D0	53	10	20	06	31	04	06	25	07	76	64	FA	08	72	63	107	172
30E0	23	10	06	FA	05	23	08	72	54	23	33	09	53	43	34	172	
Sun	03	43	2F	37	31	50	B3	FA	75	76	6F	70	46	3F	39	4F	170

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15sum	
3300	40	51	10	30	06	31	45	13	33	07	61	51	02	35	76	10	D	
3310	64	43	31	07	52	31	13	23	05	52	44	13	33	06	44	25	E	
3320	16	25	05	63	53	13	33	06	72	36	33	05	61	51	13	12	A	
3330	33	06	72	61	02	35	05	76	64	13	23	06	63	53	02	35	14	B
3340	07	72	36	FA	03	44	35	13	23	04	75	31	02	35	05	14	B	
3350	63	53	06	25	06	76	64	10	20	07	31	04	FA	06	71	10	A	
3360	52	10	20	04	66	46	17	37	03	75	42	35	44	04	55	44	15	0
3370	06	25	05	71	52	04	26	01	71	52	15	35	02	61	51	06	1E	5
3380	25	03	72	61	13	33	04	64	54	11	22	05	FA	06	72	12	12	E
3390	06	63	43	FA	06	76	65	34	08	44	08	55	63	22	03	09	18	B
3400	52	64	12	22	10	62	02	03	24	11	42	32	05	14	26	44	10	9
3410	52	65	43	15	31	41	07	05	14	63	51	45	54	16	40	40	19	2
3420	25	46	13	52	73	07	05	14	72	62	17	37	12	51	41	16	13	F
3430	16	13	64	52	35	41	64	52	05	14	12	71	70	07	05	15	B	
3440	13	65	45	11	31	13	45	14	23	12	42	33	22	33	13	1F	2	
3450	72	62	14	23	12	60	40	35	45	13	42	32	07	05	12	65	14	1
Sum	5E	DB	F5	F7	BE	09	01	F2	92	E9	C3	5A	B1	7C	B2	F6	1E	9

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	16um
3400	45	17	27	13	42	32	16	36	10	42	42	33	42	09	52	73	14B
3410	12	22	10	62	42	05	23	11	42	32	23	12	09	65	45	12	18F
3420	32	10	43	32	05	52	11	52	73	10	24	10	52	64	32	42	125
3430	10	52	73	65	24	09	71	70	03	24	08	55	76	05	03	07	12D
3440	76	64	02	03	08	65	53	13	02	09	67	47	07	06	10	72	1FC
3450	73	34	44	11	64	45	16	26	12	74	65	05	41	13	60	50	1D5
3460	41	14	12	75	64	44	55	13	64	05	45	13	66	55	10	19C	
3470	20	12	74	65	05	41	13	60	50	41	14	14	52	40	14	11	16A
3480	14	45	57	17	27	14	73	72	17	27	12	73	72	22	14	12	164
3490	75	31	05	41	11	55	45	17	27	10	43	54	22	34	11	72	135
34A0	73	12	22	15	44	35	44	25	14	64	45	05	23	10	52	40	14F
34B0	05	41	11	61	52	41	23	10	74	65	10	20	11	72	73	05	182
34C0	23	10	52	31	10	20	11	43	54	22	34	10	74	56	34	44	116
34D0	48	43	34	22	34	04	45	12	22	08	72	73	13	02	09	1C6	
34E0	43	34	22	34	10	64	45	12	22	11	67	47	54	46	11	65	169
34F0	55	05	23	09	66	56	07	06	10	75	66	34	44	09	67	47	169
Sum	A7	AE	13	F3	1A	72	5F	32	CC	9B	28	EB	E7	26	7B	A3	117

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3500	07	06	08	45	55	07	06	09	43	24	34	10	64	45	12	17D	
3510	22	09	67	47	16	10	43	34	22	34	09	52	31	04	44	1F6	
3520	10	55	25	22	14	09	73	34	44	10	55	45	13	02	07	10C	
3530	75	31	05	23	08	52	64	34	44	09	67	47	22	03	10	31	121
3540	13	04	13	09	31	22	13	22	10	62	42	16	36	11	67	47	17A
3550	17	27	09	62	42	53	02	08	43	24	34	09	31	13	04	84	186
3560	13	08	76	64	10	20	07	43	34	22	34	08	72	73	12	22	11A
3570	09	65	53	34	15	10	54	44	35	44	10	76	57	01	00	10	11B
3580	75	53	05	23	09	76	64	05	41	09	76	55	05	23	10	55	17A
3590	43	04	14	10	75	64	04	10	55	44	23	34	08	76	55	11F	
35A0	05	41	07	65	45	44	08	55	22	03	09	76	64	05	45	12E	
35B0	41	10	60	50	41	30	11	67	57	17	27	09	52	73	03	24	17A
35C0	10	62	42	12	32	09	67	57	05	41	10	60	50	41	23	08	131
35D0	55	44	35	44	09	76	64	05	41	06	75	31	FA	07	52	13A	
35E0	04	25	03	08	65	45	34	44	09	76	57	10	20	09	64	56	17F
35F0	22	14	10	13	04	13	09	74	47	04	10	14	64	56	07	14E	
Sum	DD	9A	CA	AB	E6	13	6B	D5	6B	5D	4B	DA	DF	10	EB	95	142

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum	
3600	05	08	76	55	34	44	09	31	22	13	10	35	47	22	13	1C2		
3610	08	63	53	22	03	07	45	34	34	44	08	76	57	05	41	09	130	
3620	57	65	22	14	10	31	13	05	43	10	60	40	41	23	10	57	1E1	
3630	65	02	03	08	31	22	13	22	09	76	57	17	27	08	63	43	18C	
3640	17	27	09	65	73	16	26	08	63	53	44	33	09	62	53	22	19A	
3650	41	10	31	13	25	13	08	67	57	17	37	09	76	64	05	41	10A	
3660	06	65	45	16	36	09	76	55	44	04	48	55	47	17	27	09	1D9	
3670	66	56	16	36	09	52	64	05	14	07	66	56	33	43	08	52	173	
3680	31	10	20	07	76	44	02	03	08	67	57	13	02	07	76	57	1F6	
3690	33	08	73	05	23	06	76	55	34	44	08	73	07	76	57	18B		
36A0	FA	08	63	53	05	41	08	45	05	32	09	52	73	04	14	1C0		
36B0	08	63	43	05	41	09	60	50	41	30	08	72	73	05	23	09	13C	
36C0	63	53	44	53	07	45	33	22	03	08	54	43	FA	09	63	150		
36D0	53	05	23	09	65	34	44	09	76	57	10	20	09	64	56	17F		
36E0	05	23	06	76	64	FA	07	65	55	02	03	08	67	57	13	18B		
36F0	02	09	66	44	33	06	45	34	44	09	76	57	10	20	09	64	56	17F
Sum	80	06	24	04	45	BA	71	F2	7C	2B	15	0D	3E	28	47	8C	142	

Hex	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3700	08	66	43	33	04	09	52	73	05	41	07	52	64	05	23	08	12B
3710	63	43	04	14	07	74	56	04	15	08	72	13	10	20	07	74	140
3720	65	34	44	08	76	64	05	32	07	47	10	20	08	76	57	18B	
3730	05	32	06	75	63	43	06	54	34	43	07	52	31	43	16D		
3740	53	08	62	53	22	41	05	65	45	34	44	06	FA	05	32	1CB	
3750	07	67	57	17	37	08	74	47	22	14	09	47	65	02	13	08E	
3760	52	40	32	41	07	74	56	04	15	08	72	35	46	08	67	17E	
3770	57	17	37	07	52	04	14	09	64	43	53	44	75	83	16F		
3780	02	13	10	76	55	FA	07	76	57	02	13	07	52	40	32	19B	
3790	14	08	63	43	04	05	07	74	47	32	14	08	63	53	02	13	16A
37A0	76	64	02	13	07	63	43	04	53	08	75	20	12	13	06	135	
37B0	66	45	35	46	07	57	17	37	06	63	43	10	20	07	76	193	
37C0	57	05	41	08	75	64	22	14	05	76	54	44	06	75	31	16C	
37D0	05	23	05	75	31	02	13	06	52	64	FA	0A	FA	0A	12	1A2	
37E0	22	05	65	45	34	45	06	56	46	56	07	74	56	01	20	199	
37F0	06	74	07	45	23	07	76	57	17	27	05	65	95	23	06	1EB	
Sum	DF	4D	B3	3F	4B	CD	2F	2F	65	CD	25	5C	60	EC	35	79	134

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3800	64	44	35	44	05	64	54	05	23	04	65	45	34	45	05	74	1A6
3810	47	05	14	03	66	56	13	33	04	65	45	34	44	05	64	54	14B
3820	12	32	06	52	31	10	20	03	65	45	34	44	04	63	53	44	120
3830	53	02	63	53	24	26	05	03	72	36	13	33	04	FA	FA	01	132
3840	04	14	02	24	06	45	05	75	FA	76	35	15	27	22	22	05	180
3850	36	14	01	14	09	44	03	10	25	05	10	05	04	04	04	64	1E0
3860	56	04	13	12	52	33	05	16	10	74	63	22	42	11	95	43	114
3870	34	43	12	52	64	12	32	10	17	47	04	15	11	75	64	23	174
3880	33	10	74	65	35	17	11	53	43	10	20	12	43	33	22	10	1F9
3890	13	75	42	10	31	14	52	31	20	31	14	65	63	31	92	15	167
3900	04	14	02	24	06	45	05	75	FA	76	35	15	80	06	15	77	177
3910	74	31	41	13	52	47	22	14	45	35	04	04	04	04	04	34	100
3920	23	34	14	55	47	01	11	72	74	28	42	12	75	42	34	1A5	
3930	44	13	77	74	04	13	14	55	63	25	35	13	43	33	22	30	15A
3940	11	66	56	04	13	12	52	33	05	16	75	66	22	10	12	1B8	
3950	63	43	34	44	13	43	33	22	14	11	72	74	05	12	16	66	157
Sum	0D	45	4E	42	C5	9E	D8	9F	E1	69	D0	C7	B8	85	9F	1D7	

チェス・プログラム・リスト

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 3D00 76 +01 +25 +06 16 12 74 63 13 21 07 45 34 23 34 08 66 157
 3D10 56 35 49 09 53 44 10 20 10 75 66 05 41 07 74 63 1AE
 3D20 17 27 08 45 34 23 34 06 66 5A FA 01 75 66 35 1E3
 3D30 44 07 27 08 45 34 23 34 06 66 5A FA 01 75 66 35 1E3
 3D40 10 52 33 05 16 09 36 54 16 36 10 72 73 04 20 1BE
 3D50 76 64 04 26 76 55 FA 07 67 57 10 20 60 60 126
 3D60 50 17 27 09 44 35 24 35 08 44 35 24 35 09 36 54 1B6
 3D70 23 33 07 60 30 10 20 60 35 24 35 08 44 35 24 35 1B6
 3D80 35 08 36 54 35 49 09 53 43 46 55 09 36 23 33 1B6
 3D90 06 76 64 17 27 07 44 35 24 35 05 64 54 02 26 06 1E0
 3DA0 72 73 FA FA 07 36 25 16 25 05 65 55 17 27 06 36 1AF
 3DB0 47 16 36 07 45 62 23 33 05 52 33 04 15 66 62 62 1E2
 3DC0 02 24 05 68 55 24 06 65 45 02 24 06 64 54 19A
 3DD0 07 36 25 16 25 08 76 55 22 14 08 72 73 34 44 105
 3DE0 09 53 43 22 14 08 74 63 23 33 09 37 35 43 44 06 126
 3DF0 45 34 23 34 06 76 55 34 44 07 56 25 16 25 08 55 113

Sum SE 81 74 69 90 53 09 10 12 F2 CF 68 48 01 56 184

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 3E00 53 44 54 08 55 47 FA FA 09 64 54 23 33 07 53 44 14B
 3E10 35 44 06 56 34 44 07 36 44 35 44 08 64 72 73 FA 1B1
 3E20 FA 08 52 44 24 60 07 56 25 16 25 08 64 54 FA 16A
 3E30 04 66 56 25 13 05 75 13 21 06 FA FA 23 33 07 15A
 3E40 44 34 44 35 48 08 55 03 05 09 53 44 33 10 52 73 108
 3E50 44 33 44 35 48 08 55 03 05 09 53 44 33 10 52 73 108
 3E60 04 31 07 65 45 34 44 08 64 54 02 13 09 53 44 33 108
 3E70 44 10 36 25 16 25 11 76 64 FA 08 53 44 35 44 1E1
 3E80 09 57 02 03 02 10 64 54 05 32 10 72 73 12 22 09 196
 3E90 76 55 02 04 34 06 76 17 27 09 36 25 14 25 13 122
 3EA0 44 33 43 48 08 53 43 FA FA 09 36 25 16 25 10 64 54 1B3
 3EB0 05 23 09 55 34 04 37 09 43 14 17 27 52 73 02 181
 3EC0 24 08 76 55 FA FA 09 36 25 16 25 09 55 34 07 157
 3ED0 07 64 54 05 14 08 65 45 16 25 07 64 44 22 08 FA 124
 3EE0 45 04 34 06 76 17 27 09 36 25 14 25 13 122
 3EF0 FA 35 44 09 53 44 02 57 06 60 40 10 30 07 57 66 116

Sum BB 85 19 CB 94 AB A5 9B 86 D2 C0 1F BC A2 5F D3 12E

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 3F00 05 14 08 67 47 17 27 09 36 34 00 20 68 67 57 04 18A
 3F10 15 09 44 35 44 06 54 13 21 06 48 FB 07 56 54 33 122
 3F20 07 36 25 16 25 08 64 00 22 09 57 66 02 24 09 16B
 3F30 65 55 23 03 06 57 66 02 24 07 60 40 FA 07 66 101
 3F40 11 00 01 08 11 66 23 33 07 65 45 17 27 08 36 47 15B
 3F50 FA 06 76 55 05 35 45 48 08 53 44 35 44 07 57 02 1E2
 3F60 27 06 45 45 35 45 48 08 53 44 35 44 07 57 02 1E2
 3F70 09 36 25 16 25 09 FA FA 23 33 05 75 66 17 27 06 116
 3F80 36 63 34 44 07 FA FA 23 33 08 64 54 11 21 09 65 1C2
 3F90 54 44 53 08 72 73 13 21 09 53 44 35 44 07 67 10B
 3FA0 23 33 08 72 73 13 21 09 53 44 35 44 07 67 10B
 3FB0 74 09 53 43 34 42 07 56 46 44 53 08 62 53 25 46 1AB
 3FC0 09 FA 13 21 07 53 45 44 08 FA 08 FA 11 21 09 17F
 3FD0 64 54 02 11 76 64 FA FA 09 65 45 02 11 76 64 1E3
 3FE0 54 FA 09 67 47 02 11 76 67 07 06 11 72 73 1E2
 3FF0 FA FA 07 52 43 21 08 61 51 02 24 07 64 54 16 17A

Sum SE 03 24 2F 31 0B 48 22 26 EC 9B 51 A5 4B 07 11B

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4000 26 06 36 25 13 25 07 68 44 35 44 08 53 44 16 36 1E2
 4010 07 52 33 05 16 09 36 54 16 36 10 72 73 04 20 1BE
 4020 64 54 05 16 07 FA FA 02 24 06 65 45 34 44 08 66 18C
 4030 11 06 01 09 11 66 23 33 08 64 54 23 33 09 53 43 19D
 4040 24 44 10 76 64 FA 06 36 54 13 21 07 60 40 34 1F5
 4050 44 05 52 33 04 15 06 76 17 27 07 36 25 13 25 16A
 4060 08 62 74 44 07 32 25 36 06 62 42 17 27 07 1E2
 4070 26 25 13 25 05 FA 05 14 04 65 01 13 05 65 1B1
 4080 24 23 34 06 64 44 35 45 07 52 33 05 14 08 76 55 12B
 4090 25 33 09 44 35 14 36 08 36 25 13 25 09 FA 25 16F
 4100 53 08 25 33 09 44 35 14 36 08 36 25 13 25 09 FA 25 16F
 4110 09 53 43 17 27 16 25 13 25 09 FA 25 13 25 09 1C2
 4120 26 47 16 36 10 26 25 13 25 09 76 55 17 27 10 36 1C4
 4130 47 16 36 07 76 55 17 27 08 36 47 16 36 09 47 65 129
 4140 11 21 08 26 25 13 25 09 FA 05 23 07 FA 17 104
 4150 27 08 36 25 13 25 09 76 16 36 08 54 16 36 16 16A

Sum 82 87 63 78 BF DC CB F2 6E 79 17 70 74 05 37 23 186

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4160 09 47 65 11 21 07 53 43 17 27 08 36 25 13 25 09 166
 4170 43 34 04 34 06 FA 17 27 07 36 63 34 44 08 53 15A
 4180 44 25 44 08 54 15 21 09 53 43 42 03 13 10 52 73 13A
 4190 64 54 05 16 07 FA FA 02 24 06 65 45 34 44 08 66 18C
 4200 26 06 36 47 34 45 07 44 74 64 08 73 05 14 123
 4210 06 45 27 36 07 24 25 13 25 06 26 13 25 07 116
 4220 54 27 34 05 52 33 04 15 06 76 45 24 34 07 64 184
 4230 44 17 27 08 45 34 23 34 06 66 5A FA 01 75 66 35 1E3
 4240 15 06 74 47 16 07 74 47 27 36 08 47 07 57 1E3
 4250 35 05 64 74 17 27 06 36 47 07 66 76 55 16 76 1DC
 4260 09 47 65 11 21 08 65 36 27 06 47 56 25 07 07 1CE
 4270 FA 16 36 08 47 65 13 21 06 36 25 13 25 05 66 1C2
 4280 24 17 27 08 45 34 23 34 06 66 5A FA 01 75 66 35 1E3
 4290 24 44 05 FA FA 17 27 06 76 55 25 13 05 64 44 17 17A
 4300 27 06 36 47 54 07 53 44 17 32 08 FA 02 13 133
 4310 09 74 42 11 31 76 42 53 31 41 09 75 53 23 37 08 14B
 4320 44 81 F2 CA 78 52 70 52 72 68 90 BE FC 2F E 181

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4200 75 42 16 76 09 74 64 04 26 10 77 74 02 24 11 FA 13A
 4210 FA 05 14 12 72 73 FA FA 07 52 44 05 16 36 34 148
 4220 35 45 07 54 63 16 36 06 36 25 14 07 54 17 1E2
 4230 32 06 36 72 35 44 07 53 44 33 08 55 63 56 36 1A4
 4240 08 61 51 04 26 06 65 54 17 27 06 36 25 13 25 05 17D
 4250 53 43 17 27 06 36 25 13 25 04 64 14 17 27 05 36 192
 4260 65 01 22 06 76 64 16 36 05 36 54 01 22 06 FA 15E
 4270 35 45 06 74 64 35 45 06 74 64 35 45 06 74 64 15E
 4280 24 06 65 45 23 33 06 76 55 01 13 07 53 43 23 102
 4290 06 74 63 01 13 07 53 43 13 01 26 53 43 23 37 07 1B8
 4300 74 63 12 22 08 76 55 01 13 09 75 64 FA 08 52 122
 4310 04 01 13 05 66 56 02 24 06 76 66 22 04 65 55 10E
 4320 02 24 05 64 54 23 33 05 74 66 54 FA 06 53 16 36 108
 4330 04 01 13 05 66 56 02 24 06 76 66 22 04 65 55 10E
 4340 04 74 63 25 13 05 66 56 17 27 04 64 54 25 13 05 156
 4350 53 43 04 37 02 65 45 34 44 03 63 53 06 25 04 53 130

Sum FA 3A 84 26 3D F1 8B 83 52 SE CB 4A 6E EE 24 18B

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4300 44 35 44 05 64 54 13 33 06 61 51 01 22 07 72 61 173
 4310 02 13 08 FA FA FA 09 72 73 22 14 10 76 57 12 118
 4320 22 11 57 65 17 37 12 66 56 14 15 13 75 57 05 179
 4330 14 74 64 04 14 15 57 35 13 35 16 67 57 02 03 17 100
 4340 35 45 07 54 63 16 36 06 36 25 14 07 54 17 1E2
 4350 35 13 35 14 60 50 41 23 15 35 13 35 16 67 57 108
 4360 04 14 17 71 60 07 06 15 71 60 04 14 16 74 63 07 1FF
 4370 03 12 52 64 01 00 13 66 56 14 15 14 64 63 35 23 173
 4380 15 62 22 32 15 77 76 05 14 13 60 40 22 32 14 143
 4390 35 43 71 70 34 13 14 34 31 35 13 35 16 67 57 108
 4400 43 33 42 14 51 42 14 35 15 64 63 35 43 17 74 63 1C5
 4410 14 35 14 62 42 05 32 12 52 40 22 32 13 61 34 01 149
 4420 11 77 76 17 37 12 57 36 14 35 13 61 72 05 41 060
 4430 13 67 57 37 47 12 57 65 37 47 11 75 64 17 37 12 1E5
 4440 52 30 14 13 62 42 22 32 11 52 64 15 13 62 64 189
 4450 43 35 23 10 67 57 17 37 11 66 56 04 15 12 76 64 1B9

Sum 85 55 A9 1F 0F 37 35 7F 31 9C 46 03 56 5C 95 9B 191

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4400 14 35 13 64 43 35 43 12 77 67 14 35 13 67 63 05 196
 4410 41 12 52 64 14 35 13 64 43 35 23 14 74 64 04 15 18B
 4420 12 77 76 10 20 73 74 65 37 47 16 36 37 47 12 1E9
 4430 60 40 14 35 13 64 56 35 23 12 74 65 15 35 13 64 189
 4440 56 35 23 12 64 43 37 07 13 74 65 10 20 14 51 41 155
 4450 52 35 13 64 43 35 12 74 65 10 20 14 51 41 155
 4460 47 14 51 41 00 35 13 51 41 37 07 14 41 31 16 26 107
 4470 11 66 56 04 15 12 76 64 14 35 10 76 64 10 20 11 146
 4480 51 41 14 35 12 52 33 25 33 10 66 56 17 37 11 75 16A
 4490 52 35 12 73 63 44 45 12 32 14 34 31 35 13 67 57 108
 4500 14 35 12 52 33 25 33 10 66 56 17 37 11 75 16A
 4510 13 61 72 04 26 12 74 63 33 43 13 54 47 44 54 12 1C3
 4520 74 65 07 05 11 72 73 22 14 12 31 13 04 13 52 1E3
 4530 40 32 14 76 64 11 37 15 40 52 10 20 14 66 56 190
 4540 11 31 14 61 34 13 24 10 76 64 05 41 11 72 73 22 16A

Sum 3F 80 51 77 27 74 72 80 CA 1C 6B F5 9C EB 4E 1AF

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4500 14 72 65 12 32 13 77 76 01 00 12 60 50 41 30 177
 4510 11 60 50 41 30 12 72 73 22 14 74 65 45 30 21 13 1AF
 4520 51 41 30 12 14 64 43 12 32 14 74 65 45 30 21 13 1AF
 4530 65 22 14 13 51 41 30 21 10 66 56 05 32 79 75 31 045
 4540 10 20 10 31 22 13 22 11 76 64 05 32 12 52 73 04 1C5
 4550 14 13 67 57 32 41 12 64 43 22 13 11 76 57 05 32 158
 4560 12 74 65 02 12 73 02 03 11 72 73 05 32 12 129
 4570 67 57 17 37 12 66 66 25 46 11 67 57 05 32 12 129
 4580 73 04 15 13 74 65 11 31 13 66 66 07 07 12 66 66 105
 4590 33 45 13 54 43 44 54 12 74 63 43 13 54 43 44 141
 4600 64 54 05 16 07 FA FA 02 24 06 65 45 34 44 08 66 18C
 4610 12 75 57 05 41 11 77 76 14 35 12 67 07 04 26 11 176
 4620 52 73 14 35 12 62 42 33 42 09 76 64 22 14 10 52 18A
 4630 73 17 37 11 77 76 14 35 12 64 43 35 25 11 73 65 05
 4640 01 00 10 67 57 17 37 09 68 54 05 41 11 60 50 41 189
 4650 42 11 75 57 22 14 57 15 43 13 51 41 30 21 13 1AF

Sum 7B 33 BE 62 2C 86 F4 02 AB 05 00 A9 74 04 BB BB 1D2

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
 4600 10 72 73 22 14 11 76 57 13 46 10 57 57 22 14 11 185
 4610 72 73 12 22 11 74 64 17 37 07 67 57 17 37 08 75 1E8
 4620 31 02 13 09 31 22 13 22 10 72 61 FA 08 72 61 189
 4630 10 20 10 31 22 13 22 11 76 64 05 32 12 52 73 04 1C5
 4640 14 06 72 61 01 22 07 FA 02 13 08 26 75 31 FA 1C4
 4650 09 57 57 17 37 10 31 22 13 22 10 72 73 FA 09 197
 4660 67 52 17 10 76 64 04 15 08 67 57 17 37 09 76 181
 4670 64 54 05 16 07 FA FA 02 24 06 65 45 34 44 08 66 18C
 4680 FA 09 57 65 17 27 06 67 57 12 32 07 66 46 33 1E5
 4690 43 05 61 51 44 54 06 74 36 05 41 07 72 61 04 19A
 4700 08 FA FA FB 07 52 71 08 08 08 56 54 41 32 07 151
 4710 52 40 06 08 62 52 58 FA 09 74 64 22 10 98 76 55 106
 4720 27 13 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34
 4730 43 08 52 35 23 05 34 66 46 25 46 03 66 46 35 15A
 4740 46 04 67 57 46 57 05 75 57 06 25 66 64 54 13 33 15A
 4750 07 57 02 02 01 61 51 21 09 72 61 02 11 10 150
 4760 02 FA 02 61 80 25 94 AC A1 C0 92 6F 51 C9 48 163

チェス・プログラム・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
5100	22	14	12	14	16	54	20	13	46	36	41	52	14	36	16		
5110	25	52	61	15	72	71	14	25	12	65	55	22	33	46	46	16	
5120	50	51	14	60	51	25	15	46	36	13	21	11	66	46	36	15	
5130	15	12	43	25	34	35	13	52	33	12	03	14	46	35	22	30	184
5140	15	33	14	03	14	12	65	55	22	30	13	66	46	30	51	14	1E5
5150	60	51	24	34	14	16	66	25	13	12	43	35	24	35	13	52	146
5160	33	12	03	14	46	36	13	21	11	66	46	36	15	15	15	15	15
5170	15	32	15	32	77	72	02	13	14	76	56	05	02	15	64	37	107
5180	26	34	27	14	05	13	72	71	02	13	14	76	32	51	14	2	
5190	13	67	67	11	31	14	47	37	32	51	14	65	55	02	13	15	11
51A0	64	67	41	41	13	43	22	12	22	12	67	22	54	13	15	65	176
51B0	45	34	14	03	14	12	65	55	22	30	13	66	46	30	51	14	1E5
51C0	24	34	10	43	22	11	22	11	FB	FB	06	07	10	43	31	23	1B8
51D0	33	11	44	33	14	41	08	FA	FA	FA	09	43	20	11	22	1A1	
51E0	07	42	51	10	20	08	72	54	22	30	09	65	45	11	31	10	1EF
51F0	FA	FA	03	12	11	45	35	24	12	43	66	02	11	13	52	110	
Sum	7F	29	85	01	1F	0F	DC	9E	7F	4F	E9	28	17	E3	46	19A	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um	
5600	45	34	25	37	13	44	34	25	33	14	52	33	41	33	15	54	12E	
5610	45	12	14	14	54	43	33	52	41	61	52	41	62	12	64	53	11B	
5620	31	13	17	60	06	07	14	70	73	05	01	15	45	35	31	15	1ED	
5630	41	14	76	77	05	01	13	43	22	12	22	14	44	34	23	34	1B8	
5640	15	43	34	25	37	13	44	34	25	33	14	52	33	41	33	15	12E	
5650	45	34	25	37	13	44	34	25	33	14	52	33	41	33	15	54	12E	
5660	45	12	14	14	54	43	33	52	41	61	52	41	62	12	64	53	11B	
5670	45	12	14	14	54	43	33	52	41	61	52	41	62	12	64	53	11B	
5680	32	24	34	15	45	34	23	34	14	74	56	14	32	11	60	50	163	
5690	22	43	12	54	43	24	34	13	45	34	23	34	14	74	56	14	13B	
56B0	32	13	43	54	11	31	61	51	02	13	10	61	45	11	31	31	1A1	
56C0	11	60	40	FA	FA	10	60	50	FA	FA	09	64	53	05	14	10	142	
56D0	72	54	FA	FA	09	73	53	05	14	10	72	54	FA	FA	10	64	1E0	
56E0	25	54	FA	FA	09	64	55	05	14	10	72	54	FA	FA	09	43	22	1E0
56F0	12	22	10	64	55	23	33	09	45	35	05	14	10	35	24	15	146	
56D0	24	11	64	37	16	26	12	37	55	FA	08	72	54	11	31	1A1		
56E0	09	65	55	02	11	10	73	63	01	13	11	60	40	31	41	09	65	104
56F0	61	41	05	14	12	60	40	FA	FA	10	60	50	FA	FA	09	64	53	05
Sum	41	25	0E	91	95	29	70	D3	7D	C7	0E	E9	42	FC	E4	5A	1DF	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
5200	33	25	33	14	51	33	30	42	15	73	72	11	33	15	70	71	1C9
5210	42	54	15	33	11	42	54	13	60	40	41	41	52	33	30	173	
5220	42	54	13	73	55	02	14	12	64	30	02	11	13	55	56	23	1A6
5230	14	45	34	25	37	12	60	01	40	31	41	12	45	35	24	13	164
5240	52	33	25	33	14	51	33	30	42	15	73	72	11	33	15	70	1C9
5250	33	25	30	51	12	60	01	51	25	33	13	73	55	02	14	55	54
5260	05	32	14	55	00	54	66	12	62	51	25	33	13	73	55	33	1B5
5270	54	15	43	22	03	13	72	70	35	22	10	45	35	24	34	169	
5280	11	45	34	25	30	12	60	01	31	41	13	52	33	25	33	13	172
5290	73	72	11	33	15	70	71	1C9	54	15	34	25	30	12	60	01	170
52A0	54	FA	FA	10	65	45	22	43	11	54	43	11	31	12	44	34	1DB
52B0	23	33	14	35	34	25	13	14	52	46	02	11	14	73	46	02	1A7
52C0	11	12	73	55	02	11	13	60	00	30	14	70	74	31	41	17B	
52D0	12	60	90	02	11	13	73	55	20	30	14	44	34	23	34	09	1E4
52E0	45	22	11	22	11	43	55	00	51	08	FA	FA	05	14	09	72	170
52F0	45	03	30	09	FA	23	33	10	43	22	11	22	11	22	11	25	1FE
Sum	50	80	31	94	20	32	99	74	64	77	D0	61	D0	31	51	B8	12A

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
5700	45	31	41	10	44	34	41	52	11	34	25	16	25	16	25	16	1C9
5710	02	11	10	52	31	12	11	09	60	50	02	11	10	65	55	164	
5720	14	11	73	FA	FA	12	11	74	56	01	22	10	61	01	05	14	1E4
5730	11	65	55	FA	FA	09	61	41	05	14	10	65	45	FA	FA	10	141
5740	05	54	FA	FA	08	72	36	05	14	09	64	51	31	10	46	25	151
5750	73	11	31	09	65	45	FA	FA	09	65	55	FA	FA	08	65	55	105
5760	05	14	09	72	54	FA	FA	09	73	53	FA	FA	09	73	74	1B9	
5770	FA	FA	08	73	74	05	14	09	72	54	FA	FA	07	65	05	13	1B8
5780	08	72	54	05	14	09	64	56	11	31	10	46	26	25	13	11	1B3
5790	60	50	01	22	12	43	22	12	33	13	67	47	02	11	14	73	109
57A0	43	FA	FA	09	45	35	FA	FA	09	60	50	FA	FA	09	61	41	106
57B0	FA	FA	09	60	40	FA	FA	09	64	55	01	13	10	73	64	13	161
57C0	21	11	FB	FB	21	42	12	66	46	42	54	10	45	15	33	12	1A2
57D0	11	35	24	FA	FA	09	45	35	FA	FA	08	66	01	22	09	1B5	
57E0	46	36	25	13	10	45	35	05	14	11	35	24	07	05	08	64	139
57F0	55	01	22	09	72	54	05	14	10	73	64	22	43	11	54	43	154
Sum	85	6D	7E	5A	77	6D	05	99	B4	0D	D2	40	5A	6A	1E	0B	136

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
5800	02	12	61	52	04	FA	10	61	51	08	FA	FA	09	64	53	05	142
5810	22	12	61	52	04	FA	10	61	51	08	FA	FA	09	64	53	05	142
5820	07	40	50	15	04	18	72	54	FA	FA	09	62	40	22	43	10	1E0
5830	73	45	11	21	11	72	36	02	11	12	70	75	00	02	11	65	121
5840	45	23	33	14	61	41	02	20	15	74	05	02	15	72	61	186	
5850	05	02	07	FA	05	14	08	72	54	FA	FA	09	65	45	FA	FA	10
5860	37	10	44	33	24	33	11	42	64	05	14	26	77	10	50	134	
5870	06	75	64	10	20	07	FA	FA	03	12	08	65	45	01	22	09	1FD
5880	76	77	05	14	45	35	FA	FA	11	35	24	34	12	64	90		
5890	42	42	43	51	FA	01	72	36	11	14	72	66	11	14	172		
58A0	02	15	10	45	51	FA	11	64	55	11	31	12	60	50	02	170	
58B0	11	66	46	23	33	10	64	55	FA	11	66	46	22	43	105		
58C0	12	73	45	23	13	13	46	36	11	31	14	60	40	02	11	14	1C0
58D0	72	54	02	11	14	65	22	33	15	44	33	14	52	33	14	172	
58E0	47	02	11	11	60	40	02	13	12	66	46	22	43	13	73	43	10C
58F0	13	22	14	46	36	25	13	15	52	33	22	33	16	44	33	24	19D
Sum	CD	3B	CE	67	FA	9B	33	DD	20	CA	47	EF	F2	E3	F6	1C	1C0

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15u
5900	24	34	12	43	54	20	13	46	36	41	52	14	36	16			\$25
5910	FA	09	FA	25	13	14	10	72	54	FA	09	66	46	05	14	1A	\$8
5920	10	46	FA	25	13	11	45	35	24	35	12	44	FA	0B	12	1B	\$5
5930	10	46	FA	25	13	11	45	35	24	35	12	44	FA	0B	12	1B	\$5
5940	46	11	91	02	46	36	25	13	10	60	50	01	22	11	65	45	\$83
5950	00	11	09	45	35	13	14	35	24	15	24	14	FA	05	05		\$B6
5960	14	10	61	41	05	11	60	50	FA	0B	05	55	14	05	55		\$B6
5970	14	10	61	41	05	11	60	50	FA	0B	05	55	14	05	55		\$B6
5980	46	11	91	02	46	36	25	13	10	60	50	01	22	11	65	45	\$83
5990	55	05	14	10	76	17	13	05	11	73	74	05	26	12	72	54	\$7E
59A0	FA	09	FA	13	70	73	24	34	11	72	54	05	26	12	73	74	\$7E
59B0	FA	09	FA	13	70	73	24	34	11	72	54	05	26	12	73	74	\$7E
59C0	13	73	6A	FA	11	67	47	16	32	12	66	45	36	27	36	13	\$E4
59D0	72	36	25	33	10	49	35	13	13	66	46	FA	11	35	44		\$C6
59E0	24	32	14	22	43	24	15	24	15	24	13	73	74	FA	13	72	\$54
59F0	FA	13	46	FA	07	FA	0B	60	40	11	07	65	45	03	13		\$37

チェス・プログラム・リスト

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 5800 40 25 13 21 10 77 76 01 13 11 65 45 02 11 12 45 11A
 5810 35 24 35 13 43 25 13 24 10 75 64 01 13 11 65 45 11A
 5820 24 34 12 43 35 16 22 13 FB FB 05 23 05 23 09 73 63 02 192
 5830 11 10 65 55 01 21 11 FB FB 05 14 12 75 63 02 192
 5840 12 67 47 FA 12 77 76 FA 09 65 45 02 11 10 17D
 5850 45 35 31 11 35 31 52 12 24 15 65 15 13 05 18
 5860 42 23 33 09 75 53 04 14 10 67 76 FA 10 77 76 131
 5870 FA 09 75 66 02 11 08 77 76 11 31 09 65 50 02 10D
 5880 11 10 72 54 13 32 09 75 53 02 11 10 65 45 13 21 1FE
 5890 08 60 40 01 22 09 72 54 13 34 10 43 51 22 30 11 1EB
 5900 51 30 63 70 12 45 45 24 22 13 67 47 05 14 75 125
 5910 64 02 13 11 77 63 00 02 12 FB FB 22 30 11 43 51 1EB
 5920 05 14 12 75 63 FA 12 60 FA 09 67 47 05 1A
 5930 14 09 75 53 FA 09 60 40 FA 09 77 76 FA 1A
 5940 07 67 57 23 33 08 45 25 33 09 52 35 24 33 10 1E7
 5950 73 64 05 14 11 75 66 FA 09 75 66 01 22 08 13 100
 Sum BE FB IE 1B CA FA DE D4 19 09 5E 97 70 IE D1 CD 1BF

Sum BE FB IE 1B CA FA DE D4 19 09 5E 97 70 IE D1 CD 1BF
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 5C00 22 11 22 49 34 25 33 10 34 25 03 23 08 47 57 1B1
 5C10 05 14 09 45 45 FA 06 75 31 02 13 07 FA 0A 05 1B1
 5C20 08 75 66 FA 06 75 55 01 22 07 66 56 05 14 144
 5C30 FA 24 34 09 65 45 05 14 10 76 77 FA 10 72 18B
 5C40 54 FA 10 73 55 FA 09 72 54 05 14 08 FA 1A 1F6
 5C50 FA 07 0A 05 14 08 72 54 FA 06 72 54 10 1A6
 5C60 20 07 71 11 31 08 FA 0A 02 07 74 05 1A
 5C70 10 65 55 05 14 11 60 40 31 40 12 52 40 FA 08 1A5
 5C80 40 31 41 09 52 64 02 11 10 65 55 01 13 11 FA CD
 5C90 FA 05 14 12 73 FA 0A 08 60 50 12 11 09 65 45 17E
 5C00 01 13 10 73 55 05 14 FA 07 54 01 22 08 13 100
 5C10 07 60 40 11 21 08 75 53 02 11 09 FA 0A 05 1C7
 5C20 10 73 64 05 14 08 66 46 24 34 09 43 25 16 26 109
 5C30 66 26 36 25 11 44 35 03 12 07 65 55 11 31 08 73 1F2
 5C40 65 02 11 07 73 55 03 12 07 65 45 01 13 07 66 46 1D2
 Sum C9 06 01 5B BD 25 6D EC 4A AC BA 74 7F AA E7 24 173

Sum C9 06 01 5B BD 25 6D EC 4A AC BA 74 7F AA E7 24 173
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 5D00 01 22 08 46 36 25 13 09 77 76 05 14 10 67 47 FA 1A6
 5D10 FA 06 66 56 10 20 07 75 66 05 12 08 FA 0A 01 22 02
 5D20 09 52 64 02 13 10 62 52 05 14 11 67 57 FA 0A 09 1D7
 5D30 43 64 11 31 10 72 36 05 02 11 11 67 57 05 14 12 72 18B
 5D40 54 FA 11 72 36 05 02 11 11 67 57 05 14 12 72 18B
 5D50 34 42 09 61 51 05 14 10 72 61 FA 11 52 64 2A 10C
 5D60 34 09 75 74 02 13 10 43 22 11 22 11 52 40 0A 01 1B7
 5D70 12 62 42 22 13 10 40 52 05 14 14 65 45 0A 01 1B8
 5D80 52 34 31 14 12 65 45 02 13 10 73 64 05 14 12 72 18B
 5D90 34 25 34 14 73 64 FA 14 43 25 45 46 15 72 54 15C
 5DA0 46 54 16 74 12 21 17 73 64 14 32 18 70 74 FA 1D5
 5DB0 05 65 55 14 34 06 75 31 01 15 07 43 35 23 33 196
 5DC0 08 44 31 10 20 08 45 24 34 09 43 25 16 26 109
 5DD0 62 42 05 32 12 71 52 FA 13 42 31 20 13 FA 1B9
 5DE0 FA 32 43 15 76 77 02 20 15 52 44 25 44 14 55 44 152
 5DF0 15 35 15 44 44 10 13 FA 15 44 24 34 16 76 77 15 174
 Sum 54 21 00 82 29 08 1E 73 94 C3 BE D1 B4 BA 44 3B 17A

Sum 54 21 00 82 29 08 1E 73 94 C3 BE D1 B4 BA 44 3B 17A
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 5E00 25 12 FA FA 12 61 51 FA 1A 11 60 40 02 15 1A8
 5E10 12 40 31 20 31 13 70 10 11 00 14 73 64 05 32 15 19F
 5E20 71 52 FA FA 16 52 44 25 44 16 FA 2A 25 33 15 FA 13D
 5E30 FA FA 15 61 51 FA 1A 11 52 05 32 15 FA 1A 1C0
 5E40 FA 15 64 FA 1A 14 FA 1A 14 FA 02 31 71 52 FA 1E5
 5E50 FA 12 FA 05 32 13 40 31 FA FA 13 71 52 FA 199
 5E60 11 FA 05 32 12 60 40 02 11 13 40 31 20 31 14 73 1B8
 5E70 61 51 05 32 12 60 40 00 01 13 40 31 20 31 14 73 1B8
 5E80 53 FA FA 14 71 52 FA 1A 14 FA 1A 14 FA 13 23 178
 5E90 FA 13 FA 1A 14 FA 1A 14 FA 1A 14 FA 13 23 178
 5EA0 FA 12 FA FA 12 71 52 FA 1A 12 73 53 FA FA 1B9
 5EB0 12 71 52 FA 12 66 46 FA 1A 12 72 61 FA 1A 11 179
 5EC0 FA 05 32 12 61 51 FA 1A 12 73 53 FA 1A 11 71 131
 5ED0 52 02 11 12 73 53 FA 1A 12 73 53 25 33 14 34 23 15C
 5EE0 13 75 15 FA 05 14 16 75 73 33 25 12 61 51 76 7E
 5EF0 FA 17 72 54 FA 12 52 44 25 33 13 54 33 11 33 1A9
 Sum CA B4 35 61 27 6A 81 7B 7F D5 C9 5E 6B F1 EF 8B 1F2

Sum CA B4 35 61 27 6A 81 7B 7F D5 C9 5E 6B F1 EF 8B 1F2
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 5F00 14 73 64 05 14 72 54 13 22 16 73 64 FA 1A 12 66
 5F10 FA 0A 05 14 75 76 FA 1A 12 61 51 FA 1A 11 60 40 02 15 1A8
 5F20 40 11 31 10 40 51 13 21 11 35 54 05 32 12 71 52 1D7
 5F30 02 11 13 75 53 FA 13 54 35 FA 0A 09 31 64 13 121
 5F40 21 10 73 54 25 33 11 54 35 21 33 09 FA 0A 05 32 132
 5F50 10 76 77 FA 07 43 35 25 33 08 44 33 23 30 09 17E
 5F60 31 40 11 31 10 40 51 13 21 11 35 54 05 32 12 71 1D7
 5F70 52 02 11 13 75 53 FA 1A 13 73 64 FA 13 54 FA 17
 5F80 FA 13 60 40 FA 12 6A FA 1A 12 6A FA 12 73 64 FA 178
 5F90 FA 12 73 53 FA 1A 11 FA 05 32 12 76 77 FA 1A 1F5
 5FA0 11 72 54 02 11 12 73 53 21 33 23 30 02 10 11
 5FB0 FA 05 32 11 76 77 31 40 09 31 64 13 21 10 35 1B1
 5FC0 54 25 33 09 31 42 13 21 10 42 51 65 11 71 52 10A
 5FD0 FA 11 35 34 FA 0A 09 31 13 03 13 10 34 FA 05 1EE
 5FE0 32 11 76 77 FA 1A 12 52 44 25 33 13 54 33 11 33 1A9
 5FF0 11 35 54 FA 0B 72 36 05 14 09 43 33 14 10 13C
 Sum 94 34 6B C4 33 59 07 37 BA 10 B9 07 F0 F2 2B 22 147

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 6000 62 52 41 23 07 43 51 10 20 08 31 64 11 31 09 FA 1C5
 6010 FA 02 11 10 62 42 31 42 11 64 42 13 21 12 51 30 1B2
 6020 21 42 06 43 51 23 33 07 72 36 05 14 08 44 33 02 19B
 6030 33 09 71 52 14 10 14 36 63 33 03 11 75 31 01 22 10D
 6040 12 73 64 FA 12 60 50 01 52 13 61 52 FA 13 1FF
 6050 63 54 FA 12 FA 0A FA 0A 06 43 31 10 20 05 75 1C7
 6060 31 02 13 63 31 01 13 15 01 51 52 26 08 FA 1A6
 6070 05 19 72 FA 0A 12 60 FA 10 42 00 02 73 54 18B
 6080 20 12 70 73 11 31 10 65 45 10 20 11 73 63 13 34 17F
 6090 09 65 55 FA 0A 09 73 55 FA 09 73 53 FA 0A 09 1A8
 60A0 65 45 FA 0A 08 72 54 FA 0A 05 65 55 14 24 06 62 1B8
 60B0 01 22 07 71 52 05 05 14 08 72 54 FA 0A 05 1B8
 60C0 23 30 11 42 33 24 33 11 52 33 25 12 73 33 03 10B
 60D0 12 11 44 33 22 41 12 62 41 14 21 09 73 63 23 33 1C3
 60E0 10 42 33 24 33 11 43 22 11 22 12 44 34 25 04 09 1A1
 60F0 75 64 23 33 10 43 22 11 22 10 42 33 24 33 11 44 06
 Sum E5 23 C6 6E 7B B9 A0 6A CA DD BF CC F9 05 9D 65 1AE

Sum E5 23 C6 6E 7B B9 A0 6A CA DD BF CC F9 05 9D 65 1AE
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 6100 33 22 41 12 43 51 41 33 08 43 62 FA 0A 09 62 54 110
 6110 23 33 10 42 33 24 33 11 44 33 22 34 12 73 51 14 1FA
 6120 32 33 10 46 23 33 10 42 33 24 33 11 52 33 25 33 134
 6130 12 44 33 22 41 08 43 22 11 22 09 75 64 FA 0A 09 1A8
 6140 72 63 FA 09 72 54 FA 0A 09 72 54 FA 0A 05 1B8
 6150 31 0A 07 72 54 05 14 08 71 52 FA 07 75 64 1A6
 6160 05 14 08 73 FA 0A 07 73 51 05 14 08 71 52 FA 1B2
 6170 FA 06 53 24 34 07 43 64 23 33 05 75 64 14 34 1A4
 6180 06 43 35 05 14 07 71 52 05 14 08 71 52 FA 05 1B8
 6190 FA 0A 07 52 FA 0A 07 52 FA 0A 04 73 43 01 22 05 75 1D7
 61A0 31 03 13 06 43 51 10 20 07 31 22 11 22 08 FA 19C
 61B0 14 34 06 31 22 11 22 07 FA 1A 14 34 08 47 53 05 1A8
 61C0 14 09 62 42 02 10 75 73 06 25 11 72 36 FA 1A5
 61D0 08 43 05 05 14 09 75 73 06 25 10 71 52 36 14 24 06 62 1B8
 61E0 71 52 22 43 07 55 43 06 25 08 72 36 14 24 06 62 1B2
 61F0 42 22 43 07 55 43 06 25 08 71 52 14 24 06 62 54 111
 Sum 50 4D 09 EF 65 41 E4 B6 04 76 64 B2 AB 4B 5B B9 10C

Sum 50 4D 09 EF 65 41 E4 B6 04 76 64 B2 AB 4B 5B B9 10C
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 6200 10 61 41 11 02 43 51 10 20 08 31 64 11 31 09 FA 1C5
 6210 13 09 31 13 03 13 03 75 31 02 13 04 40 01 22 1F7
 6220 05 FA 06 25 06 63 15 26 07 71 63 05 16 08 111
 6230 63 42 FA 0A 07 74 10 20 10 31 22 13 22 06 63 1B6
 6240 53 16 26 06 75 74 14 24 07 62 52 10 20 08 31 75 141
 6250 14 09 63 43 32 43 52 43 52 43 33 16 64 14
 6260 24 07 75 75 05 14 08 63 43 32 43 09 55 07 12 105
 6270 10 71 52 09 62 52 43 52 10 31 61 52 FA 0A 1D4
 6280 62 42 06 25 05 71 52 13 31 06 42 31 14 74 04 73 113
 6290 11 29 09 62 54 05 14 08 63 43 32 43 09 55 07 12 105
 62A0 22 43 08 55 43 32 43 09 31 13 03 13 10 52 31 25 195
 62B0 44 07 63 43 32 43 08 55 43 22 43 09 31 13 03 13 1CE
 62C0 10 62 42 05 14 11 71 52 FA 0A 31 13 03 13 05 1F8
 62D0 62 52 06 25 06 63 55 14 24 07 FA 0A 05 14 08 73 162
 62E0 64 FA 09 53 43 22 10 75 73 64 08 75 54 155
 62F0 FA 06 73 64 12 22 07 62 53 32 43 09 71 63 42 555
 Sum 13 3B FD 82 6D 0C 56 EE C2 2B C5 6B 85 CA 79 39 19A

Sum 13 3B FD 82 6D 0C 56 EE C2 2B C5 6B 85 CA 79 39 19A
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 6300 05 05 FA 01 22 08 43 53 14 24 07 71 63 05 25 149
 6310 08 60 50 05 14 09 70 71 FA 1A 10 61 41 32 41 08 1D7
 6320 75 74 05 14 09 63 75 FA 1A 10 75 56 00 03 06 75 120
 6330 74 06 25 07 63 43 32 43 08 72 36 23 33 09 36 25 128
 6340 16 25 10 44 33 13 33 07 62 52 14 24 06 63 05 1AE
 6350 14 24 34 22 34 06 61 51 06 05 07 75 16 26 109
 6360 08 72 61 05 16 09 62 42 14 34 10 71 52 FA 0A 06 1B8
 6370 73 64 16 26 07 62 52 05 16 08 75 73 14 34 09 63 1D7
 6380 43 34 43 10 52 43 22 43 06 62 52 06 25 07 75 199
 6390 08 52 14 06 63 43 32 43 52 43 52 43 33 16 64 14
 63A0 44 07 73 64 14 24 06 63 43 05 14 09 75 73 32 43 1B7
 63B0 10 52 43 23 33 08 60 50 22 30 08 61 41 30 51 10 141
 63C0 70 50 51 72 07 65 43 25 44 09 43 33 22 01 09 75 1C8
 63D0 74 71 10 30 70 10 40 74 FA 0A 06 50 14 50 72 50 1A8
 63E0 05 16 12 73 64 FA 0A 05 14 24 07 FA 0A 05 14 08 73 162
 63F0 22 07 63 53 16 26 08 50 10 51 16 09 71 52 14 1F2
 Sum 9F 55 2B FF C2 DD 10 69 EF BE EC 19 70 71 BB DE 07 1D7

Sum 9F 55 2B FF C2 DD 10 69 EF BE EC 19 70 71 BB DE 07 1D7
 Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F +Sum
 6400 10 61 41 11 02 43 51 10 20 08 31 64 11 31 09 FA 1C5
 6410 11 14 01 14 01 63 05 14 08 50 20 43 33 42 01 22 1F7
 6420 33 22 41 10 63 53 05 14 11 55 14 08 63 43 1F1
 6430 32 43 07 53 33 FB 03 61 41 32 41 04 63 43 03 08 1B8
 6440 25 05 75 53 14 24 06 FA 0A 05 14 07 67 63 23 33 1AE
 6450 08 44 34 25 15 09 55 74 01 22 10 73 64 FA 0A 09 173
 6460 10 76 77 FA 07 43 35 25 33 08 44 33 23 30 09 17E
 6470 50 14 10 50 10 70 50 FA 07 60 50 41 50 50 72 50 1A8
 6480 FA 0A 00 50 05 25 03 61 51 01 22 04 72 61 05 1B8
 6490 25 05 71 52 16 26 06 73 01 02 17 07 FA 05 16 100
 64A0 52 34 14 06 63 43 32 43 08 55 10 63 43 09 12 106
 64B0 75 64 05 14 08 73 FA 0A 07 75 74 FA 0A 0A 0A 1E8
 64C0 52 06 25 05 72 61 16 26 06 73 01 05 16 07 52 33 1AE
 64D0 FA 03 62 42 14 34 04 71 52 15 35 05 63 53 06 1B5
 64E0 25 06 75 03 01 22 07 44 35 02 08 55 47 35 24 126
 64F0 75 64 23 33 10 43 22 11 22 10 42 33 24 33 11 44 06
 Sum EC 45 27 EA DE 12 32 B1 C8 B1 CF 7F D8 AC DF 96 1B9

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
6F00	73	53	14	24	02	65	45	01	22	03	76	55	16	26	04	63	13E
6F10	53	05	16	05	62	52	13	23	06	66	56	14	34	07	75	66	14F
6F20	06	14	08	FA	FA	FA	FA	6A	06	75	64	14	24	07	75	66	122
6F30	14	08	73	24	23	03	08	72	54	FA	FA	06	72	54	06	25	112
6F40	07	67	57	FA	07	75	64	FA	07	51	05	06	15	05	15	05	157
6F50	FA	06	25	06	31	22	11	22	07	73	64	FA	06	62	15	05	158
6F60	42	FA	06	63	53	FA	03	66	56	13	33	04	44	33	16	6	166
6F70	03	03	03	75	42	14	24	04	71	52	06	25	05	44	34	13	16A
6F80	33	06	34	25	33	02	06	42	31	25	44	03	63	53	14	24	10A
6F90	04	76	55	06	65	66	56	13	23	02	63	43	11	21	02	10	10B
7000	62	05	02	05	04	34	25	33	04	53	43	03	43	03	73	14	149
7010	43	14	24	06	75	42	01	22	07	43	54	13	23	08	76	55	1F2
7020	63	23	04	08	75	42	33	25	06	76	55	01	22	07	43	44	13A
7030	23	08	71	63	23	04	05	52	43	13	23	06	42	33	21	137	
7040	07	42	51	23	04	08	73	14	24	08	73	55	03	43	07	1F8	
7050	42	31	02	13	08	73	63	13	31	06	76	55	02	07	71	117	
Sum	91	97	7E	54	78	32	B1	90	F0	B1	3C	50	F3	1B	7E	3B	1E2

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
7400	52	31	16	27	14	43	65	25	13	45	33	03	50	12	75	11B	
7410	64	22	43	13	FA	FA	10	20	08	73	51	01	13	09	FB	10F	
7420	03	30	10	72	71	00	01	11	43	34	23	14	21	50	40	10D	
7430	40	10	66	46	00	01	11	67	57	11	31	12	42	31	22	103	
7440	10	43	34	23	34	11	51	40	30	40	12	52	40	11	21	108	
7450	43	34	23	34	09	73	03	05	03	10	72	24	11	21	41	104	
7460	51	01	20	12	72	53	15	23	08	73	62	01	15	09	FB	171	
7470	03	14	10	66	46	10	20	11	43	33	22	32	12	42	33	1177	
7480	31	07	43	34	23	34	08	73	03	05	03	09	52	33	25	33	172
7490	10	42	33	12	22	11	75	42	11	31	12	42	51	02	11	118E	
7500	70	72	10	30	14	76	64	30	40	15	33	22	11	22	14	100	
7510	50	30	40	15	50	61	41	14	60	40	14	14	20	40	15	50	14C
7520	22	33	15	51	60	22	33	14	33	22	01	22	15	33	03	108	
7530	13	13	22	01	22	14	51	33	22	41	51	33	11	41	62	185	
7540	14	76	64	32	20	15	64	72	30	42	13	FB	FB	10	30	14	186
7550	33	22	01	22	15	73	03	00	15	33	03	02	12	33	12	33	1E6
Sum	3D	EB	C9	03	C4	2C	0E	23	34	09	12	75	08	1B	20	D8	1C3

Add	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
7600	52	23	34	08	43	34	33	41	09	73	03	04	09	60	50	10B	
7610	03	73	08	75	31	00	01	07	75	31	03	30	08	71	52	33	107
7620	52	07	75	51	14	24	08	75	31	02	13	08	71	52	33	52	168
7630	07	34	23	03	23	08	71	52	02	07	75	64	16	26	08	10B	
7640	FA	05	16	04	04	34	25	33	04	10	55	34	FA	07	75	100	
7650	42	33	21	08	42	31	23	33	08	42	51	23	34	09	43	108	
7660	22	30	05	76	55	01	22	05	75	42	03	10	66	73	64	33	126
7670	21	07	42	53	01	22	07	42	51	43	53	08	64	44	12	22	1F4
7680	06	75	51	12	34	07	76	64	43	53	06	73	43	53	21	04	19B
7690	76	55	01	22	05	71	50	16	26	06	46	56	05	16	07	75	149
7700	66	33	12	05	63	43	32	43	06	52	45	13	23	04	75	42	157
7710	31	05	42	51	13	33	03	75	53	01	22	04	76	76	55	32	121
7720	42	05	53	03	13	33	06	44	34	25	13	07	73	64	14	24	10E
7730	04	73	05	33	03	03	63	53	16	26	01	63	43	06	25	02	17C
7740	62	42	16	26	03	71	52	05	16	04	44	13	25	05	65	100	
7750	55	FA	06	72	54	14	34	07	47	33	12	22	08	76	64	1F0	
Sum	41	D6	3E	C4	F4	F0	44	3C	FF	53	77	E0	D2	F4	71	B6	112

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum	
7800	52	22	31	42	13	22	14	03	53	11	70	73	22	33	12	44	33	104
7810	11	21	13	17	66	04	01	20	14	64	52	20	09	52	31	01	1FB	
7820	10	31	10	31	10	01	41	09	70	73	03	73	10	74	73	02	24	151
7830	11	52	31	01	20	12	60	50	12	22	13	31	52	16	05	08	1BD	
7840	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	1BD
7850	73	03	14	12	61	51	05	03	07	75	53	25	48	08	54	63	1AF	
7860	34	43	09	52	31	46	34	10	62	12	32	12	31	1F	FB	10	1BD	
7870	22	12	31	50	01	22	07	73	63	34	43	08	54	43	01	22	1EC	
7880	09	43	54	22	01	61	51	10	32	20	11	73	25	13	10	1F4		
7890	08	FB	22	34	11	12	35	05	32	43	63	65	07	07	07	07	100	
7900	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	100	
7910	75	25	14	75	25	14	75	25	14	75	25	14	75	25	14	75	17B	
7920	15	05	09	72	54	03	47	10	54	65	47	36	11	66	56	01	177	
7930	26	26	72	36	12	32	07	43	33	14	24	08	73	63	24	33	1FC	
7940	09	52	53	02	24	10	76	64	24	33	11	42	33	01	13	11	180	
7950	44	33	05	04	12	FB	FB	10	10	75	53	24	31	51	61	54	180	
7960	31	01	13	09	27	66	27	16	27	66	27	25	44	11	55	44	197	
Sum	AA	B0	87	63	DE	9E	AB	45	1F	96	F6	59	7A	90	69	05	1FC	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum		
7600	05	04	09	36	27	16	07	11	67	57	27	25	44	09	66	46	03	12B	
7610	30	10	36	27	16	07	11	67	47	27	47	09	66	57	03	73	10E		
7620	05	04	09	36	27	16	07	11	67	57	27	25	44	09	66	46	03	12B	
7630	76	64	13	34	11	FA	FA	17	27	07	43	32	03	01	73	10E	1E4		
7640	63	22	06	76	44	34	05	03	10	63	72	25	11	31	55	45	109		
7650	01	22	06	76	44	14	34	07	72	36	12	22	08	73	63	03	10B		
7660	13	09	43	33	03	21	10	36	27	16	07	11	67	57	27	25	44	03	12B
7670	05	04	09	36	27	16	07	11	67	57	27	25	44	09	66	46	03	12B	
7680	75	53	14	34	07	43	32	33	27	08	76	64	15	35	09	72	190		
7690	54	03	47	10	54	65	47	36	11	66	56	01	20	05	76	55	1E2		
7700	FA	06	75	44	14	34	07	FA	FA	01	22	09	43	33	22	08	109		
7710	14	09	25	15	10	54	65	47	36	11	66	56	01	20	05	76	55	1E2	
7720	05	04	09	36	27	16	07	11	67	57	27	25	44	09	66	46	03	12B	
7730	32	12	13	25	15	42	32	26	34	10	72	14	20	15	12	15	147		
7740	32	12	13	25	15	42	32	26	34	10	72	14	20	15	12	15	147		
7750	13	20	16	32	23	20	31	56	31	50	36	14	24	13	22	23	199		
7760	13	20	16	32	23	20	31	56	31	50	36	14	24	13	22	23	199		
Sum	9B	CF	94	24	D2	FF	9F	B6	2A	9E	14	26	D4	30	A5	115	115		

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum
7700	46	13	72	26	16	46	46	13	25	12	72	54	12	72	54	12	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B300	63	45	17	27	14	70	22	03	12	63	05	03	21	12	75	166	
B310	74	10	20	10	67	57	46	55	07	43	32	03	08	36	63	160	
B320	30	32	09	76	35	02	40	10	FA	FA	46	55	11	64	55	01	188
B330	22	10	63	54	32	20	09	63	23	07	44	34	05	03	107		
B340	10	63	54	25	46	11	64	46	02	46	12	67	57	46	24	08	177
B350	73	40	30	32	09	36	54	32	22	10	40	62	02	24	11	70	135
B360	72	24	42	06	66	46	12	32	07	43	33	14	24	08	46	36	07
B370	25	04	09	72	54	24	33	09	67	47	24	33	10	42	33	02	184
B380	13	11	47	37	11	21	11	72	54	14	34	07	43	33	01	13	189
B390	08	66	46	46	15	09	67	47	15	10	46	46	35	26	35	08	108
B400	67	47	23	32	09	73	62	25	10	64	46	02	46	11	65	184	
B410	55	46	13	12	67	37	15	05	66	56	FA	FA	06	75	66	11E	
B420	14	34	07	72	64	34	08	64	43	01	22	09	43	22	11	1F1	
B430	22	10	FA	FA	15	11	73	62	03	25	12	56	64	05	04	13D	
B440	10	44	54	01	04	09	43	62	02	14	01	61	03	13	11	14C	
B450	FA	FA	24	57	12	65	57	66	13	76	66	25	37	10	32	1A5	
Sum	5A	EB	7E	4D	0F	43	D3	05	EA	BE	60	3E	5A	FB	5E	EA	1B2

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B800	36	25	03	25	12	75	13	22	10	61	51	12	32	11	43	10B	
B810	31	10	20	12	31	50	02	35	13	62	73	23	33	14	72	36	25
B820	33	10	63	13	64	44	20	44	13	62	63	25	33	14	72	61	33
B830	43	10	62	53	21	13	08	61	51	05	04	09	73	62	34	45	5E4
B840	10	55	63	13	32	11	72	50	12	22	09	67	57	17	27	10	109
B850	43	34	23	34	11	72	61	44	09	72	61	12	22	10	73	18D	
B860	62	34	09	09	73	62	11	21	08	64	54	05	04	09	61	51	16D
B870	12	22	10	72	61	34	44	11	55	63	25	33	14	72	61	34	1C7
B880	43	11	65	33	10	20	09	67	57	12	22	09	43	34	21	14	100
B890	10	72	54	03	14	11	73	63	13	32	12	75	73	10	30	11	164
B900	73	72	04	04	12	75	73	13	12	11	53	25	04	10	62	12	17F
B910	32	04	12	09	72	54	34	44	10	55	63	23	33	08	43	34	12C
B920	33	34	09	73	62	12	22	10	75	73	03	14	11	64	44	25	156
B930	42	11	65	33	10	20	09	67	57	12	22	09	43	34	21	14	100
B940	07	67	57	12	14	08	64	04	12	22	09	72	54	34	03	07	13E
B950	72	45	12	22	08	FA	FA	03	30	09	71	14	34	07	64	187	
Sum	3C	51	D8	E2	2A	3B	9B	4A	4F	4B	87	A1	13	1E	5B	72	17B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B410	33	22	34	11	62	54	12	22	12	33	25	16	25	13	73	64	113
B420	05	70	07	43	12	32	08	76	64	10	20	60	40	01	180		
B430	13	10	FA	FA	15	11	73	62	03	25	12	56	64	05	04	13D	
B440	11	67	57	25	04	12	72	54	00	01	13	70	50	04	12	10	1CA
B450	72	67	17	27	11	76	63	00	01	12	73	62	27	37	13	32	13B
B460	73	25	17	09	FA	FA	01	13	10	65	45	34	11	56	45	19F	
B470	05	04	12	60	40	25	46	15	73	55	13	35	08	67	47	01	100
B480	20	09	72	29	12	19	73	63	11	31	07	43	34	24	34	100	
B490	08	73	05	07	09	72	36	12	22	10	76	55	03	04	07	154	
B500	76	55	01	22	08	43	33	22	03	09	67	57	12	32	10	67	153
B510	05	12	09	72	54	02	46	10	FA	FA	05	04	06	76	55	161	
B520	02	46	07	67	57	46	55	08	73	55	01	22	09	43	32	13C	
B530	43	10	55	75	13	11	72	54	14	34	10	54	04	06	76	55	161
B540	11	65	45	14	34	09	55	25	13	10	72	54	14	34	11	11B	
B550	43	33	22	45	09	72	54	25	13	10	55	73	14	34	11	43	156
B560	33	22	43	12	75	66	12	32	13	FA	FA	15	35	07	75	64	1FA
Sum	05	0C	5A	AC	C0	DA	49	4B	49	1E	4C	8B	0E	EC	59	F5	1CB

add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum	
B900	44	14	34	08	FA	FA	12	32	05	72	FA	FA	06	67	57	140		
B910	12	22	07	64	54	10	20	08	75	64	11	31	09	FA	FA	31	174	
B920	42	11	65	33	10	20	09	67	57	12	22	09	43	34	21	14	100	
B930	06	70	07	12	13	07	73	63	05	04	06	73	63	05	04	07	1CE	
B940	67	57	01	22	05	64	54	FA	FA	06	75	64	01	13	07	FA	186	
B950	FA	FA	14	34	08	61	51	05	04	09	72	54	34	03	10	55	43	
B960	13	32	09	73	62	12	22	10	72	50	16	05	09	43	34	23	1E7	
B970	44	14	34	08	FA	FA	12	32	05	72	FA	FA	06	67	57	140		
B980	08	43	24	34	09	73	62	12	22	08	73	62	05	04	09	17	107	
B990	43	34	23	34	09	75	73	03	14	08	61	41	05	04	09	60	1F2	
B000	40	34	03	10	54	43	12	32	10	55	43	12	32	09	72	61	16A	
B010	34	10	54	43	10	55	07	61	41	14	08	61	41	05	04	09	60	1F2
B020	43	09	55	43	12	32	06	75	63	12	32	07	FA	01	21	55	15B	
B030	05	72	36	FA	06	64	54	12	32	07	75	63	03	30	08	1BE		
B040	FA	02	46	07	43	33	01	20	06	73	63	12	32	07	43	144		
B050	32	23	07	43	33	03	30	04	64	54	FA	FA	05	75	53	184		
Sum	79	E1	26	D7	A7	EC	CF	2A	D9	69	4D	2E	D0	60	A9	13	1B3	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B560	25	13	08	FA	01	22	09	72	54	14	34	07	75	66	25	175	
B570	13	08	72	54	14	34	09	43	33	13	35	05	67	57	01	13	1C9
B580	06	72	54	14	34	07	43	33	13	32	08	73	62	FA	FA	09	180
B590	66	46	12	22	10	76	64	22	33	10	61	41	22	33	09	61	190
B600	41	32	44	10	52	44	05	24	05	72	54	25	46	06	54	15	190
B610	14	34	07	64	FA	FA	72	45	FA	FA	06	76	57	63	12	184	
B620	32	07	43	33	04	05	04	05	76	64	FA	FA	06	72	36	17	177
B630	07	36	54	25	46	06	66	46	11	21	06	65	55	12	22	06	1EA
B640	54	56	12	32	07	43	33	22	32	08	72	54	03	30	07	43	11A
B650	33	14	24	08	33	24	15	24	09	75	64	01	22	10	FA	FA	10C
B660	03	14	08	72	53	24	33	05	64	24	33	12	32	10	73	14	105
B670	21	10	FA	02	20	10	60	40	05	04	10	67	47	17	17	10	100
B680	10	72	36	10	30	40	44	33	10	20	60	40	05	04	10	171	
B690	72	45	11	31	10	72	36	17	27	05	75	53	FA	FA	06	76	12E
B700	64	12	32	07	FA	FA	29	08	43	33	22	09	69	45	45	14D	
B710	34	53	10	73	53	25	04	07	43	32	23	08	44	34	25	1FC	
Sum	07	20	B3	C5	6F	3F	65	D8	49	F0	D9	16	10	CF	4B	9B	151

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B700	15	23	06	76	64	14	34	07	FA	FA	12	22	08	43	34	23	12F
B710	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B720	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B730	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B740	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B750	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B760	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B770	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B780	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B790	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7A0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7B0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7C0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7D0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7E0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7F0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B700	15	23	06	76	64	14	34	07	FA	FA	12	22	08	43	34	23	12F
B710	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B720	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B730	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B740	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B750	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B760	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B770	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B780	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B790	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7A0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7B0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7C0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7D0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7E0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	
B7F0	05	72	54	14	34	07	43	33	13	35	05	67	57	01	13	1C9	

チェス・プログラム・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
BD00	63	53	FA	FA	64	63	34	33	05	55	05	41	06	66	16	17	17A
BD01	56	22	54	07	73	51	41	52	06	72	36	17	27	07	67	17A	
BD20	41	52	08	61	52	13	23	09	65	55	22	34	10	64	44	34	189
BD30	26	11	47	65	FA	FA	47	42	22	12	08	36	63	FA	FA	108	
BD40	09	60	50	41	52	06	43	62	41	52	13	23	09	65	55	22	189
BD50	64	54	02	24	09	60	50	10	72	61	FA	FA	06	64	45	15	
BD60	54	FA	FA	07	73	51	41	52	06	72	36	17	27	07	67	17A	
BD70	13	22	04	64	54	05	41	05	73	62	FA	FA	06	63	55	106	
BD80	04	07	72	63	13	23	08	36	42	05	39	FA	FA	34	14	103	
BD90	06	52	33	05	41	05	73	62	FA	FA	06	63	55	23	05	25	103
BD9A	36	44	25	03	08	36	44	25	03	08	36	44	25	03	08	36	44
BD9B	44	25	03	08	36	44	25	03	08	36	44	25	03	08	36	44	25
BD9C	10	30	67	60	41	05	08	63	55	13	23	07	75	53	16	116	
BD9D	26	08	60	50	41	05	09	63	55	13	23	07	75	53	16	116	
BD9E	08	33	25	16	25	06	44	25	03	08	36	44	25	03	08	36	44
BD9F	75	64	05	04	07	FA	13	23	06	75	53	16	25	06	44	25	103
Sum	2B	39	6C	D3	F6	56	B2	4E	6C	25	08	07	73	BE	9B	15	17B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
BE00	52	13	23	05	52	33	34	44	06	33	41	22	41	07	35	43	106
BE10	12	32	04	63	53	05	41	05	72	63	FA	FA	06	64	54	13	1E3
BE20	33	06	66	56	05	04	06	52	33	41	63	05	72	36	17	27	118
BE30	04	64	44	05	32	05	32	05	34	22	34	06	63	43	41	04	1E3A
BE40	60	50	10	33	05	43	63	43	03	66	56	16	26	04	75	18C	
BE50	66	05	16	05	63	53	13	23	06	44	06	35	06	25	07	FA	169
BE60	FA	08	70	71	10	30	67	60	50	41	05	08	67	57	25	10A	
BE70	13	08	70	71	10	30	67	60	50	41	05	08	67	57	25	10A	
BE80	08	FA	FA	FA	07	76	55	06	25	08	FA	FA	FA	06	1E3		
BE90	72	63	06	25	06	76	57	06	25	06	64	04	16	10	76	143	
BE9A	64	FA	08	FA	FA	15	35	07	65	45	02	24	05	70	71	15B	
BE9B	06	25	06	61	41	FA	FA	64	54	13	23	07	75	64	FA	106	
BE9C	FA	03	64	54	16	04	64	53	23	02	64	54	16	04	56	15B	
BE9D	03	60	50	12	32	04	63	43	43	03	76	55	34	44	02	160	
BE9E	63	53	06	25	03	76	55	05	41	02	66	56	26	05	03	75	156
BE9F	66	12	22	04	76	55	34	44	04	63	43	34	23	02	76	55	10F
Sum	18	4A	E4	F3	D6	4A	2B	19	0D	8B	2C	50	E2	F6	F4	12	14D

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
BF00	34	03	61	55	43	01	22	04	63	42	06	25	04	43	22	13	18A
BF10	22	04	64	54	06	25	01	76	55	13	33	02	62	42	33	43	137
BF20	03	61	41	16	26	04	54	14	34	05	43	34	44	04	104		
BF30	72	61	05	16	03	67	53	16	26	04	54	14	34	05	43	34	190
BF40	44	12	32	04	62	33	01	22	02	64	56	12	35	06	76	145	
BF50	01	22	04	FA	14	24	05	63	53	06	25	06	71	63	05	118	
BF60	14	07	64	FA	08	75	74	03	12	08	73	64	11	31	102		
BF70	08	44	34	25	06	62	61	51	06	25	03	72	61	16	26	04	1E0
BF80	64	54	02	24	09	60	50	10	72	61	FA	FA	06	64	45	15	
BF90	64	14	24	06	FA	05	16	07	63	43	FA	02	61	41	1FA		
F00	25	03	72	61	14	24	04	60	50	12	32	01	65	45	13	1EF	
F01	33	02	76	55	06	25	03	64	54	16	26	04	75	64	05	16	11A
F02	05	04	54	16	04	54	16	04	54	16	04	54	16	04	54	16	11E
F03	42	06	25	04	64	54	13	33	01	61	41	14	34	02	72	61	12F
BF9F	15	25	00	00	00	00	00	00	00	00	00	00	00	00	00	00	13A
Sum	ED	91	4F	50	D7	4F	7B	D2	9E	F5	9A	CF	13	01	F2	03	195

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um	
9000	34	02	13	FE	20	CA	AA	90	34	03	13	FE	20	CA	AA	90	1D7	
9010	36	04	13	FE	20	C2	90	34	02	13	FE	20	C3	13	AF	18A		
9020	CD	0C	BE	7E	FE	00	C2	AA	90	FD	75	02	FD	36	01	13D		
9030	41	3E	F0	8F	6F	7E	FE	01	2B	10	3E	F0	8F	6F	7E	FE	1FA	
9040	41	3E	F0	8F	6F	7E	FE	01	2B	10	3E	F0	8F	6F	7E	FE	181	
9050	36	05	13	FE	20	C2	90	34	02	13	FE	20	CA	AA	90	171		
9060	FE	49	D2	04	CD	BE	08	DE	B5	F3	03	13	67	3A	04	19A		
9070	13	6F	CD	0C	BE	C8	7E	20	BE	F1	7D	70	BE	FA	C2	AA	1C6	
9080	90	37	3F	CF	90	75	02	7E	FD	77	03	7D	C6	F0	E6	F0	141	
9090	4F	F1	8B	9C	90	75	02	7E	FD	77	03	7D	C6	F0	E6	F0	157	
909A	41	37	3F	CF	90	00	00	00	00	00	00	00	00	00	00	00	CB	180
909B	C2	AA	90	34	02	13	CD	0C	BE	CF	BF	CF	F7	FD	77	08	1A9	
909C	CD	0C	90	C9	BD	32	CD	08	18	0D	00	93	CD	90	C9	C9	1BF	
909D	C9	00	00	00	3A	03	13	67	3A	04	13	6F	FD	36	07	00	17A	
909E	CD	0C	AB	90	76	04	13	67	3A	05	13	6F	FD	36	07	00	172	
909F	AB	90	03	03	03	03	03	03	03	00	00	00	00	00	00	00	15A	
Sum	11	2E	49	CB	3F	90	5B	6F	70	AO	C7	F6	21	9D	90	FA	1FE	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9100	3E	01	32	AB	80	21	00	43	AC	80	21	00	40	31	00	100	
9110	20	3A	0A	91	CD	00	93	CD	01	93	CD	00	93	CD	00	100	
9120	00	00	CD	AA	91	CD	00	93	CD	01	93	CD	00	93	CD	00	1E4
9130	91	CD	00	91	CD	00	91	CD	00	91	CD	00	91	CD	00	1C1	
9140	E5	B5	C5	21	28	D0	11	00	00	01	7B	00	ED	B0	C1	D1	121
9150	E1	C9	05	01	01	00	20	2E	20	44	01	54	01	4A	3A	183	
9160	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	1C8	
9170	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	174	
9180	11	52	91	CD	56	99	FA	AB	CD	60	99	11	F2	90	CD	1AB	
9190	56	99	E5	DD	E5	E1	CD	65	99	E1	F2	90	CD	40	91	CD	157
919A	E5	2A	8C	80	CD	65	E1	C9	CD	40	91	CD	40	91	CD	157	
919B	C9	E5	2A	8C	80	CD	65	E1	C9	CD	40	91	CD	40	91	CD	157
919C	E5	2A	8C	80	CD	65	E1	C9	CD	40	91	CD	40	91	CD	157	
919D	7A	AB	80	CD	77	00	CA	01	27	32	AB	80	C9	00	00	12D	
919E	E5	2A	8C	80	CD	65	E1	C9	CD	40	91	CD	40	91	CD	157	
919F	DD	23	DD	23	DD	23	DD	23	DD	23	DD	23	DD	23	DD	1C9	
Sum	EB	ED	32	87	7E	1D	4E	37	CD	1B	09	31	93	0A	12	19	18B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15um
9200	2D	20	2D	20	2D	20	2D	20	2D	20	20	00	00	00	00	00	12B
9210	3A	AB	80	D6	01	27	32	AB	80	CD	0C	92	FD	E5	DD	E5	1E3
9220	FD	21	10	10	AD	2A	AC	80	DD	7E	04	FD	77	00	77	00	160
9230	05	FD	77	01	DD	7E	04	FD	77	02	DD	7E	07	FD	77	03	12A
9240	00	AC	FD	21	00	A1	DD	7E	00	FD	77	00	DD	7E	01	16A	
9250	FD	77	01	DD	7E	02	FD	77	02	DD	7E	03	FD	77	03	CD	1E5
9260	00	AC	DD	01	AD	E1	C9	00	FD	36	06	FD	7A	CC	3A	18B	
9270	02	1E	FE	49	CA	CC	FE	58	CA	93	FE	20	7A	85	7A	85	191
9280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
9290	02	6A	AF	CD	30	BF	C3	68	92	7E	FA	18	02	3E	FB	FD	109
92A0	77	00	FD	77	01	FD	77	02	FD	77	03	C9	00	00	00	00	1D9
92B0	CD	00	92	08	2A	CD	10	92	31	00	00	C3	22	91	00	00	100
92C0	E5	06	08	0B	AC	80	2B	10	FD	22	AC	00	C1	E1	00	00	1FA
92D0	77	02	FD	00	00	00	77	03	C9	00	AF	32	AD	BD	20	00	135
92E0	05	CD	00	C5	CD	00	93	CD	30	93	18	FE	00	00	00	00	177


```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9700 3A 0B 01 23 05 21 03 02 22 11 CD 60 99 E1 C9 00 11E
9710 3E 01 32 AB 80 32 AE B0 E5 21 03 22 AC 80 21 1A4
9720 00 00 22 AC 80 22 A6 B0 21 03 22 AC 21 B0 E1 CD 135
9730 00 C6 C9 00 00 00 CD 20 AA 18 AB CD 01 A9 18 A6 11E
9740 CD 8E AC C3 E6 00 00 00 00 00 00 00 00 00 00 126
9750 7D FD BE 16 28 03 F1 37 CB FD 77 02 AF C3 F0 CB 10A
9760 CD 40 00 CD 5E 0E C9 00 00 00 00 00 00 00 00 00 13C
9770 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
9780 E5 D5 C5 C3 07 B1 00 00 E5 D5 C5 C3 B0 00 00 183
9790 11 BE B6 CD B0 07 D2 10 B3 3E 0C FD B6 FA FD 77 103
97A0 FA D2 10 B5 FD 3A FB C3 10 B3 00 00 00 00 00 00 141
97B0 11 BE B6 CD B0 07 D2 10 B3 3E 0C FD B6 FA FD 77 114
97C0 11 97 B0 CD B0 07 D2 10 B3 3E 18 FD B6 FA FD 77 114
97D0 FA D2 10 B5 FD 3A FB C3 10 B3 00 00 00 00 00 00 141
97E0 11 97 B0 CD B0 07 D2 10 B3 3E C9 97 00 00 00 00 153
97F0 FE FA CA D9 AB FE FB CA E2 AB DD 7E 01 C3 F2 AB 152

```

Sum AA 32 73 0F A9 71 20 69 FD 32 1A 5D E0 ED CF 5 135

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9800 F5 3A 0B 01 23 05 21 03 02 22 11 CD 60 99 E1 C9 00 11E
9810 AB F1 D2 10 B5 FD 3A FB C3 10 B3 00 00 00 00 00 00 159
9820 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
9830 00 00 00 00 00 00 00 00 00 00 00 00 00 10 00 11D
9840 E5 2A 7E 9B B6 32 7E 95 23 22 3E 9B E1 C9 00 00 175
9850 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9860 F5 3A 0B 01 23 05 21 03 02 22 11 CD 60 99 E1 C9 00 11E
9870 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
9880 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
9890 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
98F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum 7A 8F 66 DD 47 10 E7 0C 5D 3C 90 08 5F 65 18 91 134

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9900 7E FD 3A 0B 01 23 05 21 03 02 22 11 CD 60 99 E1 C9 00 11E
9910 C5 E5 CD 29 0C 3E 02 00 00 00 00 00 00 00 00 18A
9920 00 FE 1F 20 08 3A 00 00 32 0E CD 50 0C 18 04 114
9930 7E 1F 18 FA CD 32 08 B7 20 FA 01 00 02 CD 32 08 14B
9940 87 20 07 08 7B 81 20 F5 18 D4 F5 3A 00 00 00 00 16C
9950 0F E1 F1 C1 18 1C CD B9 08 18 17 CD C6 08 08 12 125
9960 CD DD 05 19 0D 00 B6 05 19 08 CD A4 06 18 03 CD 1FD
9970 2E 0A F5 D8 0B CB FF C3 E8 F1 C9 C5 05 11 00 9E 178
9980 21 07 9E CB 77 20 06 CB 7F 28 06 18 01 EB C3 73 1C9
9990 9B 21 0D 9D B1 C1 C9 00 00 00 C5 E5 0E 14 05 FA
99A0 06 02 7E CD B8 99 23 13 10 FB 01 28 08 09 E5 11A
99B0 2B 00 19 EB E1 18 EB D1 C1 C9 00 00 00 00 00 149
99C0 00 00 00 00 00 00 CD 99 12 CD BA 99 C9 F5 D8 0B 10F
99D0 CB FF C3 E8 3E 02 D3 FA F1 C9 00 00 00 00 00 10F
99E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 10F
99F0 00 00 99 87 20 04 3E 01 18 09 3E 2B 05 6F 3E 00 186

```

Sum 2E 9E C1 B1 E6 7A 4A 09 0D BA AE 20 E9 C2 7B 51 18A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9A00 BC 67 AF 32 0E 99 3A 0B 01 23 05 21 03 02 22 11 CD 60 99 E1 C9 00 11E
9A10 C9 0C 10 DA C1 C9 C5 D5 3E 0C B2 9A 0E 0B 168
9A20 CD B2 9A 21 C7 B9 01 BB 00 AF 77 23 08 7B 81 20 1C1
9A30 FB 36 0D 01 CB 83 0E 14 E5 11 C7 9A ED 53 B0 9A 11A
9A40 06 11 CD 7E 9A 23 10 FA 3E 18 CD B2 9A 3E BB CD 129
9A50 C0 AB AF C0 AB 21 C7 9A 11 89 00 7E 23 CD CD 196
9A60 AB 18 7A B3 20 FA E1 11 40 01 19 0C 32 9A AF 1A2
9A70 CD B2 9A 3E 0A CD B2 9A E1 B1 C1 C9 C5 D5 E5 01 136
9A80 2B 00 11 BF 9A 0E BE 05 08 7E 12 13 09 08 20 10D
9A90 F7 E1 AF ED 5B BD 9A 0E 21 BF 9A 06 08 CB E1 1A0
9AA0 1F 23 10 FA 13 0D C2 0E 00 00 00 00 00 00 00 00 1E1
9AB0 C9 C9 F5 3E 18 CD B2 9A 0E 00 00 00 00 00 00 00 1EA
9AC0 02 02 02 02 02 02 02 02 02 02 02 02 02 02 156
9AD0 2D 2F 2F 2C 3B 00 00 FF 00 3B 3C 3E 3F 3F 3F 19C
9AE0 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 1FF
9AF0 1F 1F 1F 1E 0C 00 00 FF 00 3C 3B 3C 3F 3F 1FF

```

Sum E1 3D 4A F7 5E 4A 43 D6 02 62 CA B3 B4 A6 A0 183

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9B00 3F 3C 3B 38 00 00 00 FF 00 01 03 37 37 37 37 159
9B10 37 37 3F 03 01 00 00 FF 00 0E 1F 1F 1F 1F 175
9B20 1F 1F 1E 0C 00 00 FF 00 3B 3C 3E 3F 3F 3F 134
9B30 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 183
9B40 2D 2F 2F 2C 3B 00 00 FF 00 00 00 00 00 00 00 1B3
9B50 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9B60 C5 E5 21 00 E0 01 40 1F AF 77 23 08 7B 81 20 FB 14B
9B70 E1 C1 C9 47 3A A0 B4 28 01 E8 11 2B 08 07 1F 1C7
9B80 3B 03 19 18 FA D1 C1 C9 3E 01 32 AB B0 C3 C6 113
9B90 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BA0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BB0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BC0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BD0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BE0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8
9BF0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 1F8

```

Sum DF A1 07 1B CB 6A B1 B2 15 EE C9 87 14 2E 5A 4A 19A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9C00 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C10 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C20 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C30 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C40 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C50 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C60 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C70 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C80 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9C90 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CA0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CB0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CC0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CD0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CE0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9CF0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8

```

Sum 00 F0 00 F0 00 F0 00 F0 00 F0 00 F0 00 F0 00 F0 180

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9D00 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D10 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D20 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D30 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D40 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D50 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D60 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D70 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D80 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9D90 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9DA0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9DB0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9DC0 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 1F8
9DD0 FF FF 01 01 01 01 01 01 01 01 01 01 01 01 105
9DE0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 108
9DF0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 104

```

Sum 01 F2 03 F3 03 F3 03 F3 02 F3 02 F3 02 F3 1A9

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9E00 FF 01 01 01 01 01 01 01 01 01 01 01 01 01 120
9E10 11 08 11 08 21 04 21 04 11 08 09 10 05 20 178
9E20 FD 3F 01 01 01 01 01 01 FF FF 01 01 01 01 02 107 189
9E30 21 08 51 10 09 10 05 10 05 10 05 10 05 10 10 110
9E40 21 20 11 20 09 20 05 20 05 20 05 20 05 20 10 110
9E50 FF 01 01 01 01 01 01 01 01 01 01 01 01 01 01 120
9E60 11 08 21 04 01 02 01 02 01 02 01 02 01 01 09 10 15E
9E70 05 20 FD 3F 01 01 01 01 01 FF FF 01 01 01 01 02 32 1E1
9E80 79 1E 91 09 11 08 11 08 11 08 11 08 11 08 11 08 127
9E90 11 08 01 08 09 10 05 20 FD 3F 01 01 01 01 01 01 108
9EA0 FF 01 01 01 01 01 01 01 01 01 01 01 01 01 01 108
9EB0 01 02 01 02 05 01 02 01 02 01 02 01 02 01 01 09 10 18A
9EC0 05 20 FD 3F 01 01 01 01 01 FF FF 01 01 01 01 01 01 39 18A
9ED0 A3 45 81 82 09 40 A5 25 A4 A5 A5 A5 21 84 03 40 126
9EE0 05 20 09 10 11 08 B1 11 08 B1 11 08 B1 11 08 01 14E
9EF0 FF FF 01 01 01 01 01 01 01 01 01 01 01 01 01 01 107

```

Sum DA 40 B0 62 10 EE 9C 4F 4A 3D FB 6C 5C 14 6E 2F 1DD

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
9F00 F1 0F F1 0F E1 07 C1 03 C1 03 E1 07 F1 0F 1F 170
9F10 F9 1F 01 00 01 01 00 00 FF FF 01 01 01 01 02 C1 07 185
9F20 E1 0F F1 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 172
9F30 E1 3F F1 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 01 00 13E
9F40 FF FF 81 01 C1 03 B1 01 C1 03 E1 07 F1 0F 1F 3F 1F4
9F50 F1 03 C1 03 C1 03 E1 07 F1 0F 1F 0F 1F 3F 3F 1F4
9F60 F9 1F 01 00 01 01 01 01 FF FF 01 01 01 01 02 CD 33 11A
9F70 F9 1F 71 0E F1 0F F1 0F F1 0F F1 0F 1F 0F 1F 0F 1F4
9F80 F1 0F 31 0C 1F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 01 00 1F4
9F90 FF FF 01 01 01 01 01 01 01 01 01 01 01 01 01 01 181
9FA0 F1 03 C1 03 07 F1 0F 03 C1 03 C1 03 E1 07 F1 0F 1F 18C
9FB0 FD 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 01 00 181
9FC0 A3 45 F9 1F 0F 1F 0F 1F 0F 1F 0F 1F 0F 1F 0F 1F 18F
9FD0 FD 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 3F 01 00 18F
9FE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
9FF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum DC 90 6A 8B 92 6F BA BC 7B 8A D6 B1 2E BE 80 CF 16B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
A000 48 42 40 50 40 44 42 48 40 40 40 40 40 40 14C
A010 41 41 41 41 41 41 41 41 41 41 41 41 41 41 1C7
A020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A030 00 00 00 00 00 00 00 00 41 00 00 00 00 00 00 100
A040 B1 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A060 00 B1 81 B1 81 B1 81 B1 81 00 00 00 00 00 00 187
A070 88 82 84 90 A0 A4 82 88 00 00 00 00 00 00 00 14C
A080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
A0F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum 92 8A BA A2 C2 BA 8A 92 00 00 00 00 00 00 00 00 1AB

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ISum
A100	00	00	00	00	00	00	00	FF	00	B2	00	00	00	00	00	00	00
A110	10	17	31	37	00	01	27	00	00	00	00	00	00	00	00	17	17
A120	40	B1	31	00	02	02	00	AB	00	00	00	00	00	00	00	76	55
A130	00	42	15	00	03	07	00	AB	00	00	00	00	00	00	00	20	41
A140	11	01	02	50	04	00	00	00	00	00	00	00	00	00	00	00	165
A150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	106
A170	00	00	00	00	07	07	00	00	00	00	00	00	00	00	00	00	00
A180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	108
A190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A1A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A1B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A1C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	10C
A1D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A1E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	10E
A1F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Sum	85	B5	75	70	78	00	7C	00	B2	00	0B	07	FB	08	BE	CE	100

[illegible][illegible][illegible][illegible][illegible][illegible]

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+18m
AB00	D0	2A	AF	80	B0	7E	00	FE	FF	CB	00	00	00	00	3A	AF	18m
AB10	B0	7E	00	2B	12	00	2D	23	DD	23	BB	22	77	00	2D	B2	132
AB20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
AB30	FD	7E	72	02	DD	0E	22	0D	7E	03	FD	77	00	00	7E	13E	
AB40	0A	FD	77	02	00	23	23	23	23	23	23	23	23	23	22	17F	
AB50	04	B0	C7	C9	00	00	3A	AB	B0	C6	01	27	32	AB	00	22	169
AB60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
AB70	FD	7E	02	DD	77	02	FD	7E	03	77	00	00	23	23	23	AB	148
AB80	7D	23	00	22	22	22	22	22	22	22	22	22	22	22	22	22	22
AB90	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
ABAA	C3	23	C7	20	24	AC	B0	65	0D	E5	E1	11	F4	07	37	150	
ABBA	3F	ED	52	D1	E4	B4	65	0D	C9	00	00	00	00	00	00	00	00
ABCB	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
ABD0	7E	FE	06	28	28	0B	CD	62	05	0A	00	18	F2	F1	1E	18A	
ABE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
ABF0	05	CA	00	00	18	F0	AF	07	FE	C9	F5	3E	2D	CD	CA	AB	1AB
ABFA	F1	C9	F5	3E	00	CD	CA	AB	F1	00	00	00	00	00	3E	0A	1B75
Sum	B4	E4	FE	51	AB	4D	57	E4	83	B4	D4	D9	2B	00	91	1D	1B5

[illegible][illegible]

プログラ

[illegible]

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+15
AC00	C2	20	AC	60	60	AA	C2	AC	00	00	00	00	00	00	00	00	15
AC10	92	60	60	AB	AC	D2	68	C2	00	AC	D2	AB	C9	00	00	00	181
AC20	D2	7E	03	FE	00	C2	7B	AC	00	00	00	00	00	00	00	00	186
AC30	92	60	60	AB	C2	14	AC	D2	D5	AB	C9	D2	7E	00	E6	0F	1C0
AC40	92	60	60	AB	C2	AB	D2	7E	02	AC	60	AB	C2	14	AC	C9	18D
AC50	AF	FD	00	00	00	00	00	00	00	00	00	00	00	00	00	00	190
AC60	90	00	60	79	72	AE	B9	C2	AC	00	FE	4D	C9	00	C2	98	172
AC70	AC	3E	05	C2	58	99	11	93	10	D2	A4	06	C2	80	AC	B9	190
AC80	E5	C5	06	27	21	93	10	7E	00	00	00	00	00	00	00	00	181
AC90	F6	C2	F2	AB	C1	E1	C9	00	E5	C5	06	28	21	00	00	7A	1C7
AC00	92	60	60	AB	C2	14	AC	D2	D5	AB	C9	D2	7E	00	E6	0F	189
AC10	3A	AE	80	FE	01	C2	C9	00	00	00	00	00	00	00	00	00	199
AC20	79	A9	C2	73	A9	01	C5	05	A9	C1	D5	A9	00	00	00	00	159
AC30	79	93	C2	50	A9	C0	70	A9	C0	F0	AC	00	00	00	00	00	194
AC40	70	FD	BE	16	28	03	F1	37	C9	F1	37	F3	C9	00	00	00	19A
AC50	00	28	D2	28	D2	28	D2	28	C9	00	00	00	00	00	00	00	1E9
Sum	15	59	83	C0	DF	6A	61	E9	D0	6E	24	86	02	FB	14	16	15

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+7	Isun	
AD36	FD	36	06	00	44	75	06	FE	FA	2A	6D	FE	FB	CA	2B	AD	CA	2B	13
AD37	AD	FD	7E	07	07	FE	FA	CA	26	AD	FE	FB	CA	2B	AD	CA	2B	13	
AD38	AD	FD	7E	07	07	00	00	00	00	AD	18	03	00	00	AD	CA	2B	13	
AD39	BD	CB	0D	42	72	34	00	00	00	00	00	00	00	00	00	00	00	10	
AD40	FD	CB	0A	46	72	34	00	00	00	00	00	00	00	00	00	00	00	10	
AD50	04	0A	42	07	00	3E	4B	32	05	0A	3E	60	32	06	0A	C9	19	19	
AD51	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD52	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD53	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD54	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD55	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD56	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD57	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD58	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD59	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD60	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD61	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD62	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD63	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD64	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD65	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD66	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD67	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD68	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD69	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD70	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD71	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD72	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	
AD73	04	0A	42	07	00	32	03	00	3E	48	32	02	02	0A	3E	60	32	01	

add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1 Sum
AE00	F0	02	00	00	FE	FA	CA	20	AE	FE	FB	CA	28	AE	CD	42	C7 17F
AE10	C9	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1C9
AE20	CD	40	AE	1B	0A	00	00	00	00	00	00	00	00	00	00	00	1CD
AE30	80	C9	00	00	00	00	00	00	00	00	00	00	00	00	00	00	179
AE40	FD	C8	04	46	CA	72	AE	00	3A	00	80	FE	00	20	12	AF	32 17F
AE50	AF	32	06	40	3E	4B	32	07	40	3E	60	3E	60	3E	40	C9	119
AE60	AF	32	06	40	3E	4B	32	07	40	3E	60	3E	60	3E	40	C9	119
AE70	A0	C9	3A	00	80	FE	00	20	12	AF	32	7A	00	3A	00	80	105
AE80	3E	B8	72	77	40	3E	40	32	7A	00	C9	AF	32	72	40	3E	105
AE90	71	A0	3E	B8	72	77	40	3E	40	32	7A	00	C9	00	00	00	105
AEA0	02	AF	46	46	CA	D2	AE	3A	00	80	FE	00	20	12	AF	32 17F	
AEB0	02	AF	46	46	CA	D2	AE	3A	00	80	FE	00	20	12	AF	32 17F	
AEC0	AF	32	04	A0	32	05	00	20	12	AF	32	75	40	3A	00	80	105
AED0	A0	C9	7A	00	80	FE	00	20	12	AF	32	72	40	32	73	A0	15B
AEE0	3E	B8	72	70	3E	40	32	7A	00	C9	AF	32	7A	00	3A	00	110
AEF0	75	A0	3E	B8	72	77	40	3E	40	32	7A	00	C9	00	00	00	110
Sum	43	A3	74	22	3C	EE	CC	22	EE	A3	74	22	3C	EE	CC	22	105

[illegible]

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum
B000	D0	05	E5	D5	00	21	60	21	00	00	D0	7E	00	0F	05	0F	1A0
B010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B020	D0	23	D9	23	18	E5	00	00	00	00	00	00	00	00	00	00	16C
B030	C0	7B	99	C0	9A	99	E1	23	13	13	10	C2	F7	C4	73	3A	118
B040	A0	40	A0	47	28	01	E8	11	28	00	78	1F	38	06	19	18	FA
B050	C9	7E	E1	C9	50	93	4E	86	42	86	52	87	51	2A	48	85	96
B060	CD	E3	00	00	10	0B	E4	20	2B	E2	00	00	00	00	00	00	00
B070	F3	50	68	F6	A0	48	97	00	00	00	00	00	00	00	00	00	00
B080	21	1F	0E	12	F2	0E	DF	E1	00	11	0F	EF	F1	00	01	01	28
B090	F0	0F	00	10	01	11	0F	0F	F1	FF	00	00	00	00	04	74	10C
B0A0	00	02	03	03	00	00	04	00	01	00	50	02	08	A3	02	00	18C
B0B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B0C0	E5	05	E5	3C	3A	9F	00	00	00	00	00	00	00	00	00	00	00
B0D0	BE	90	0E	0E	48	C0	5A	B1	11	93	80	0E	52	C0	5A	B1	1107
B0E0	93	90	0E	0E	60	C0	B1	11	80	B0	0E	40	00	91	11	11	CF
B0F0	7D	0E	0E	41	C0	B1	C1	D1	E1	37	3F	C9	00	00	00	00	1EC
Sum	4D	4D	50	50	1D	1E	1E	3E	3E	83	C0	80	49	49	5D	EF	156

Bdd	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+S
B100	E5	D5	C5	3A	7E	60	A7	11	89	B0	E8	04	CD	5A	B1	11	53B
B110	E8	00	0E	08	CD	5A	B1	11	93	B0	E8	04	CD	5A	B1	11	B7
B120	93	80	0E	0A	CD	0A	B1	11	80	B0	E8	02	CD	0A	B1	11	14F
B130	7A	00	0E	01	CD	0A	B1	C1	D1	E1	37	3F	09	00	00	00	129
B140	E5	1A	FE	0D	28	12	85	4F	7E	89	20	04	E1	13	18	18	197
B150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B160	97	78	B5	1A	73	B1	CD	5F	20	09	7E	FE	00	28	F1	69	
B170	89	28	DD	E1	13	C5	5A	B1	81	00	00	00	00	00	00	00	131
B180	FD	36	0C	00	FD	36	0D	80	2E	77	FD	CD	04	46	28	10	0A
B190	CD	00	B1	38	18	FD	36	05	00	CD	76	24	20	B2	20	FA	07
B1A0	E0	01	CD	5B	CA	B1	70	D6	08	4F	C3	91	81	81	81	EE	
B1B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B1C0	00	CD	7E	74	04	B2	20	FA	E0	81	CD	5B	CA	C1	B1	70	198
B1D0	06	0D	6F	C3	C1	B1	C1	B1	C3	A6	96	81	C3	C1	B1	00	164
B1E0	FD	00	0E	7E	CD	FD	05	46	28	16	E5	D5	E5	D1	00	0A	
B1F0	00	11	0E	19	10	FD	7D	75	0C	74	0D	C1	D1	E1	D1	192	
Sum	36	F8	92	93	08	C6	B6	EB	CB	11	E3	32	9D	83	D8	72	159

add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	rsum	
B200	C9	B2	00	40	A0	70	2B	35	38	E3	E5	D5	C5	21	00	11	00	C6
B210	14	C3	F4	71	00	00	00	3E	03	F6	B6	F4	77	F4	77	F4	00	171
B220	F0	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75
B230	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75
B240	F0	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75	75
B250	C1	B3	0F	04	7D	0B	93	0F	3A	00	4F	0F	04	E0	97	0F	00	1A1
B260	00	11	95	B0	E5	14	FE	0E	00	C4	B2	85	F4	A7	B2	85	00	1C0
B270	00	11	95	B0	E5	14	FE	0E	00	C4	B2	85	F4	A7	B2	85	00	1C0
B280	32	9E	B0	F4	72	75	75	75	75	75	75	75	75	75	75	75	75	75
B290	F0	C8	05	8E	F6	3A	0A	00	F6	3A	08	00	C5	10	00	C1	9A	1
B2A0	C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B2B0	F0	C8	05	8E	F6	3A	0A	00	F6	3A	08	00	C5	10	00	C1	9A	1
B2C0	7E	03	77	E1	13	C3	64	B2	05	C3	64	B2	05	C3	64	B2	05	7E
B2D0	F4	3A	F8	F7	75	75	75	75	75	75	75	75	75	75	75	75	75	75
B2E0	F7	02	F7	02	F7	03	71	00	C0	B0	30	A3	C3	B8	B2	85	00	1CF
B2F0	E1	71	7D	C8	04	46	20	04	32	9F	00	C3	9E	00	C3	9E	00	1CF

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ΣSum
B300	C9	11	93	B0	C3	10	B8	11	89	B0	C3	10	B3	11	BE	B0	1C2
B310	E5	1A	4F	FE	0D	2B	44	85	FA	73	83	CB	02	70	72	77	1EA
B320	05	77	07	77	07	07	07	07	07	07	07	07	07	07	07	07	17C
B330	03	FD	7E	11	77	CD	00	B1	38	47	FD	5A	0A	00	FD	36	143
B340	0B	00	FD	CB	05	BE	C5	10	41	C1	CD	00	B7	3E	00	FD	132
B350	BE	03	C2	6F	83	36	00	79	13	87	1B	83	E1	7D	7E	07	1B5
B360	C9	FD	CB	05	BE	C5	10	41	72	19	82	30	03	FD	3A	07	1B5
B370	05	77	07	77	07	07	07	07	07	07	07	07	07	07	07	07	17C
B380	7E	FD	77	03	FD	7E	11	77	CD	00	B1	38	47	C1	CD	00	1E1
B390	05	E5	FD	09	09	09	09	09	09	09	09	09	09	09	09	09	1AB
B3A0	CD	48	B7	3E	7E	FD	BE	03	C2	6F	83	36	00	79	C3	17	135
B3B0	83	6F	FD	CB	0A	46	2A	7B	83	C3	25	B3	00	11	71	C9	135
B3C0	05	77	07	77	07	07	07	07	07	07	07	07	07	07	07	07	17C
B3D0	20	2F	6F	FD	CB	0A	46	29	8D	CB	76	2A	24	FD	77	02	1D0
B3E0	7E	FD	77	03	71	CD	00	B1	38	46	FD	5A	0A	00	FD	36	142
B3F0	0B	00	FD	CB	05	BE	C5	10	21	C1	CD	00	B7	7D	03	4F	143

ADD	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	18uB
B400	77	E1	13	C4	B5	C8	7E	20	F7	F8	77	02	7E	77	02	77	F7
B410	03	71	C0	B0	38	19	D7	E1	D1	E4	B5	D5	E5	D9	7F	09	D4
B420	C0	B0	91	D9	19	D7	E1	D1	E4	B5	D5	E5	D9	7F	09	D4	78
B430	F7	C8	05	05	C2	F7	B3	00	C0	F7	B2	F7	00	00	20	13	EC
B440	FD	75	06	3E	6D	00	84	B4	F7	3A	06	F7	00	E1	E5	3E	10
B450	C0	B4	E1	3E	0F	00	B5	3E	11	C0	4D	B5	3E	41	09	78	00
B460	C9	70	FE	B0	38	16	E5	C6	F0	77	FE	00	20	13	EC	00	00
B470	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B480	A0	B4	E1	3E	EF	00	CB	3E	F1	C0	00	B6	3E	81	C9	E1	70
B490	85	6F	7E	FE	00	C0	C0	B6	4A	46	C4	B6	4A	0E	41	70	89
B4A0	7B	00	70	00	00	00	00	41	FD	36	00	00	00	00	00	00	00
B4B0	1B	3E	5E	FD	00	00	00	00	00	00	00	00	00	00	00	00	00
B4C0	13	C1	C0	B0	00	00	B7	3E	00	3A	50	3E	00	3A	07	5E	00
B4D0	90	C3	B0	85	E5	D9	FD	09	D9	C0	B1	D9	00	00	00	00	00
B4E0	D9	E1	D1	C1	C0	86	D8	F7	3A	00	C9	00	00	00	00	00	00

Asst	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Bun	
B500	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B500
B510	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B510
B520	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B520
B530	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B530
B540	E5	B5	6F	CB	7E	CA	80	85	4E	FD	77	02	FE	70	30	21	C4	B540
B550	7E	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	B550
B560	36	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B560
B570	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B570
B580	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B580
B590	B1	D9	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	B590
B5A0	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B5A0
B5B0	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B5B0
B5C0	FD	77	02	FE	08	3F	64	75	47	4B	4F	3A	4B	4C	4D	4E	4F	B5C0
B5D0	70	C6	FD	6F	36	B1	E1	36	00	E1	C9	D5	E5	FD	09	FD	09	B5D0
B5E0	70	C6	FD	6F	36	B1	E1	36	00	E1	C9	D5	E5	FD	09	FD	09	B5E0
B5F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B5F0
Bun	3A	4B	09	AE	EB	FA	C4	C8	3B	A7	47	B0	83	00	00	00	00	Bun

[illegible]

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+15um
8700	E5	CD	CD	CD	68	7F	0F	5E	0A	FD	0A	09	1B	FD	6E	0C	15um
8710	FD	66	00	00	7C	72	CF	0F	00	7C	7A	62	7A	62	7F	3F	FD
8720	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
8730	7D	7E	02	FD	7F	02	0F	E1	C9	C8	7A	6A	24	37	37	3F	FD
8740	ED	52	16	37	FD	D1	E1	C9	E5	05	CD	60	87	FD	5E	1C	A4
8750	FD	56	10	3F	7E	ED	52	03	05	07	00	00	00	00	00	00	A4
8760	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	A4
8770	DA	06	96	1F	DA	06	96	1F	DA	70	96	33	B2	0A	07	21	5F
8780	07	08	C9	21	41	00	C9	21	C2	00	C9	21	C2	00	C9	21	7C
8790	44	01	C9	21	B6	02	C9	B3	3F	C8	B6	07	00	B2	37	C9	1A
87A0	1F	50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1A
87B0	87	BC	7A	64	00	87	BC	7A	64	00	70	95	CB	30	0F	FD	A2
87C0	BC	04	46	20	04	E1	D1	37	C9	E1	D1	37	C9	3F	FD	CB	A4
87D0	04	46	20	05	E1	D1	37	C9	E1	D1	37	C9	3F	3E	20	C9	19
87E0	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	A4
87F0	03	C9	1F	30	3E	05	03	C9	1F	30	3E	05	03	C9	1F	30	A4

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	1Sum	
B800	FD	7E	07	FE	00	20	04	21	00	00	C9	F1	FA	D3	B8	1F	1500	
B810	DA	27	88	F1	DA	27	B8	F1	DA	28	B8	F1	DA	28	B8	1F	16E	
B820	00	68	C9	21	40	00	C9	21	40	00	C9	F1	40	01	C9	21	1F1	
B830	00	68	C9	21	40	00	C9	21	40	00	C9	F1	40	01	C9	21	1F1	
B840	21	F7	40	7E	FE	60	F5	C0	B8	F1	FE	60	AC	C5	B8	1A5		
B850	22	F2	43	B8	00	00	00	FD	21	00	A1	11	10	00	06	20	120	
B860	3E	00	FD	77	04	3C	FD	19	10	F8	FD	21	10	A1	00	00	18D	
B870	3E	00	FD	77	04	3C	FD	19	10	F8	FD	21	10	A1	00	00	18D	
B880	7D	32	9E	B8	00	7D	32	9F	B8	C9	F1	00	00	00	00	00	1BF	
B890	FD	36	0C	B0	2E	77	FD	36	0C	46	CA	6B	C3	A6	B8	1F	19F	
B8A0	B8	00	68	C9	21	40	00	68	C9	21	40	00	B9	2D	F8	C5	CA	133
B8B0	B8	00	68	C9	21	40	00	68	C9	21	40	00	B9	2D	F8	C5	CA	133
B8C0	FD	36	0C	B0	2E	77	FD	36	7E	CA	00	BC	2D	F8	CB	5D	CA	144
B8D0	C6	B8	7D	D6	08	E6	FC	CA	B8	00	00	00	00	00	00	00	189	
B8E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
B8F0	00	00	00	00	00	00	00	00	00	21	1F	C2	12	FE	DF	E1	100	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sun
B900	FD	75	00	7E	1F	4A	54	B8	1F	DA	89	49	1F	04	04	BA	11A
B910	1F	1A	84	BA	89	1F	DA	54	BA	11	F8	85	00	E5	1A	85	E5
B920	FA	76	89	89	5F	20	E1	6F	CB	76	76	89	4E	C5	10	10	1CB
B930	89	89	89	89	89	89	E1	6F	CB	76	76	89	4E	C5	10	10	1CB
B940	32	9E	80	3A	60	3A	60	E5	D9	F0	D9	C0	8B	D9	C0	8B	D9
B950	19	D9	81	00	C1	00	C1	71	79	C0	30	BB	E1	1C	C2	10	89
B960	60	7D	32	9E	80	3A	60	00	00	11	F8	85	00	E5	1A	85	1A
B970	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89
B980	0F	7D	32	9E	80	3A	60	E1	1C	C2	60	89	36	C2	C9	00	18A
B990	76	42	05	E5	D9	FD	09	00	00	00	8B	D9	FD	19	D9	00	10B
B9A0	D1	C1	71	79	C0	30	BB	E1	1C	C2	60	89	36	C2	C9	00	15A
B9B0	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89	89
B9C0	2F	20	1A	6F	CB	76	C2	D7	89	4E	C5	10	15	C1	79	CD	104
B9D0	5F	BB	CB	79	CA	BA	89	E1	1C	F2	89	36	76	48	C9	00	177
B9E0	00	09	5A	4B	E5	D9	FD	09	00	00	8B	D9	FD	19	D9	00	124
B9F0	D9	E1	C1	71	79	C0	30	BB	E1	1C	F2	89	36	76	48	C9	15A

AdA	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+10
BA00	00	00	00	00	11	11	11	11	11	11	11	11	11	11	11	11	11
BA10	55	55	14	6F	7C	7E	C2	2B	8A	45	C5	10	15	15	C1	79	CD
BA20	93	BB	CB	79	CA	5A	E1	1C	CE	59	BA	36	44	39	C9	1E	12
BA30	00	00	36	44	45	D5	D9	D9	D9	D9	CD	CD	BB	D9	1F	12	12
BA40	D9	E1	D1	C1	71	79	CD	30	BB	CB	79	CA	5A	BA	18	D7	1A
BA50	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E
BA60	55	55	20	14	6F	7C	7E	C2	2B	8A	45	C5	10	15	C1	79	CD
BA70	93	BB	CB	79	CA	5A	E1	1C	CE	59	BA	36	44	39	C9	1E	12
BA80	00	00	36	44	45	D5	D9	D9	D9	D9	CD	CD	BB	D9	1F	12	12
BA90	D9	E1	D1	C1	71	79	CD	30	BB	CB	79	CA	5A	BA	18	D7	1A
BA9A	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E	7E
BA9B	33	10	CD	00	D0	E1	3E	10	CD	D0	BA	E1	C9	00	00	14	14
BAC0	CD	00	BB	E1	3E	11	E5	CD	00	BB	E1	36	41	C9	00	00	14
BAE0	85	6F	7E	FE	00	C0	C5	10	70	C1	CD	BB	CB	79	D9	D9	1D
BAE6	36	41	D5	E5	D9	D9	D9	D9	CD	CD	BB	D9	1F	C9	D9	E1	D0
BAF0	D1	C1	7E	00	77	CD	58	BB	CB	00	00	00	00	00	00	00	1F
Sum	65	F9	62	72	38	AC	AE	BA	47	4B	51	B4	36	C3	2A	E2	17

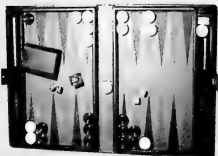
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+10
BB00	85	6F	48	7E	C8	4E	4E	10	0B	81	79	CD	98	98	98	98	15
BB10	00	36	41	05	E5	D9	D9	09	09	09	09	09	09	09	09	09	1F
BB20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1F
BB30	7E	00	CA	58	B8	1F	DA	56	B8	1F	DA	56	B8	1F	DA	56	1F
BB40	B8	1F	DA	89	B8	1F	DA	89	B8	1F	DA	89	B8	1F	DA	89	1F
BB50	C1	C6	14	C3	F8	B8	02	FE	96	1C	FD	FE	0C	CB	F5	1E	1E
BB60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1F
BB70	0E	7D	75	0F	C9	F1	D0	FF	7C	7C	7E	00	00	00	00	77	0E
BB80	7D	75	0F	C9	F6	C3	58	B8	02	FE	96	C3	58	B8	02	FE	1E
BB90	C3	58	B8	00	00	CA	BB	B8	1F	DA	89	B8	1F	DA	89	B8	1F
BBA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1F
BBB0	1E	10	FC	C1	C6	14	C3	F0	B8	02	FE	96	1C	FD	FE	0C	7E
BBBC0	05	7D	8E	0C	CB	DB	FF	77	0C	7D	7E	00	00	77	0E	FE	1E
BBBD0	75	0F	C9	00	00	7E	06	C3	BB	B8	02	FE	96	C3	BB	B8	02
BBE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1F
BBF0	7D	75	FA	C3	BB	B8	00	00	00	75	FA	C3	58	B8	00	00	1A

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+10
BC00	FD	05	70	0E	1F	DA	0A	8B	1F	DA	6B	8C	1F	DA	04	0E	+11
BC10	1F	DA	0A	8B	1F	DA	54	8B	1F	DA	B5	36	40	54	8B	1F	+12
BC20	54	8B	1F	DA	0A	8B	1F	DA	0A	8B	1F	DA	0A	8B	1F	DA	+13
BC30	0F	1	79	CD	93	BB	E1	1C	2C	1D	8C	36	40	C9	00	00	+14
BC40	2C	9F	80	76	0A	05	D5	09	FD	09	D9	CD	0A	BB	D9	CD	+15
BC50	1F	DA	0A	8B	1F	D1	E1	71	CD	30	BB	E1	1C	2C	1D	8C	+16
BC60	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	7D	+17
BC70	FA	86	86	86	86	5F	20	0F	6F	6C	7E	82	86	86	4E	C5	+18
BC80	0F	1	79	CD	93	BB	E1	1C	2C	1D	8C	36	40	C9	00	00	+19
BC90	36	62	D5	09	FD	09	D9	CD	0A	BB	D9	CD	0A	BB	D9	CD	+20
BCA0	0F	1	79	CD	93	BB	E1	1C	2C	1D	8C	36	40	C9	00	00	+21
BCB0	00	00	00	00	00	11	7C	BB	36	40	E5	1A	85	FA	07	DB	+22
BCC0	5F	20	14	6F	6C	7E	82	86	86	4E	C5	15	79	CD	93	BB	+23
BCD0	93	BB	E1	1C	2C	1D	8C	36	40	C9	00	00	00	00	00	00	+24
BCE0	00	00	00	00	00	3A	9B	00	00	00	D9	CD	0A	BB	D9	CD	+25
BCF0	D9	E1	D1	E1	71	79	CD	30	BB	CB	71	CA	BA	8C	1D	8C	+26

[illegible]

Add	0	1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+10
BE00	85	6F	CB	76	68	4E	35	10	0B	C1	79	CD	98	B9	C9	00	148
BE10	00	36	B1	C1	25	09	FD	09	D9	CD	80	B8	FD	19	00	00	149
BE20	E1	D1	81	05	20	20	20	20	20	00	93	41	88	42	86	44	150
BE30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	151
BE40	97	48	2A	50	80	60	00	81	0E	62	62	84	52	08	51	00	152
BE50	48	A0	30	00	00	00	00	00	07	62	01	06	63	02	05	00	153
BE60	03	25	65	07	33	46	05	02	4A	00	00	00	00	00	00	00	154
BE70	00	32	40	00	00	00	00	00	00	30	35	30	40	36	20	00	155
BE80	37	00	06	38	00	00	00	00	00	00	00	00	00	00	01	0F	02
BE90	18	03	1F	17	12	11	11	11	30	BE	CD	56	99	C9	00	00	156
BEA0	05	00	23	33	50	20	34	C1	00	00	00	00	00	00	00	00	157
BEB0	00	0D	20	FA	19	10	F3	C1	00	E1	C9	00	00	00	00	00	158
BEC0	7C	CD	00	BE	08	67	7D	CD	00	BE	DB	46	26	A0	23	00	159
BED0	DD	E5	05	DD	1	56	86	10	DD	BE	00	28	16	DD	4F	00	160
BEF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	161
BEFF	14	6F	37	C5	A0	00	FE	01	29	09	DD	7E	01	C1	DD	DD	162

Back Gammon



■大藤謙二
解説 ■藤原博文

バックギャモンは、数千年の歴史を持っているゲームで、日本でも盤双六として「枕草子」などにも出てきます。最近また全世界的に流行して、選手権が各地で開催されています。

スゴロクとルールは似ているので大変簡単です。ただし、一番大きな違いは、多数の駒を用いるので、スゴロクと違い同じサイコロの目でもいろいろな手があり、これが戦略のバラエティーを無限に広げ、非常に奥の深いゲームにしています。

ここで、国際ルールの簡単な解説を試みますが、充分な例を示すことはとてもできないので、もっと知りたい方は、日本バックギャモン協会の刊行物を参照してください(藤原)。

名称

盤面には、位置(ポイント)を示す24個の三角形が交互に色違いで描かれています。盤の中央には、バーと呼ばれる溝があり、全体が4区画に分れています。各自にインナーボードとアウターボードがあり、インナーボード側より1~12の番号がついています(図1)。

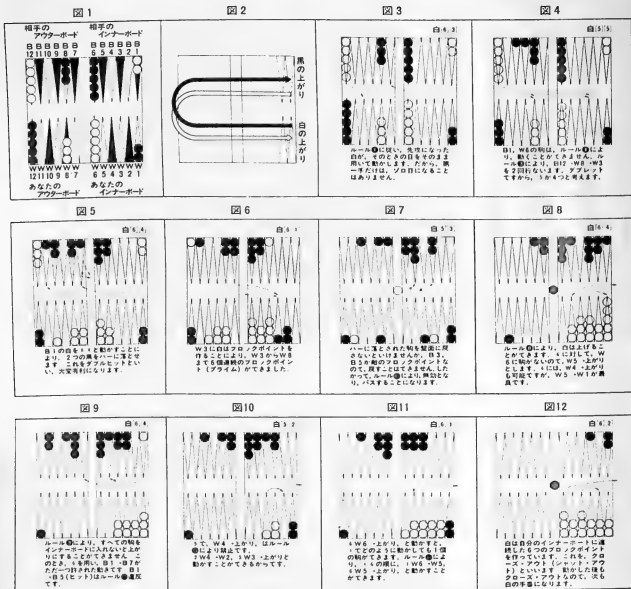
目的

ゲームの開始のときには、各自の駒を図1のように並べます。駒を図2の矢印のように、互いに逆向きに進め、15個全部を早く上げたほうの勝ちです。

とはいっても、両者の進行方向が逆であるため、すれ違いのときに、いろいろな戦略をとることにより、激戦や冷戦など様々な展開となります。

駒の動かし方

- ① ゲームの開始は、2人が1個ずつのサイコロを振り、大きい目を出した人が先攻になります。同じ目の場合は、何回でも振り直します。そして、第1手だけは、いま振ったサイコロの目をそのまま使って駒を動かします。
- ② 第2手より、2人が交互に2個のサイコロを振り、その目に従って駒を動かします。
- ③ 2個のサイコロの目が同じ場合、ダブルット(ゾロ目)といい、2つの目の2倍だけ動かします。
- ④ 自分の駒が2個以上入っているポイント(ブロックポイント)には、敵の駒は入れない。ただし、飛び越すことはできます。
- ⑤ 敵の駒が1個しかないポイントに自分の駒を進めると、敵の駒をふり出し(バー)に落とすことができ、このことをヒットといいます。
- ⑥ バーに自分の駒が1個でも残っていると、他の駒を動かすことはできません。
- ⑦ バーに落された駒は、相手のインナーボードに戻すことにより、盤上を動くことができるようになります。
- ⑧ 15個全部の駒を自分のインナーボードに入れると、駒を上がりにすることができます。このとき、サイコロの目と同じポイントの駒を取り除きます。上げるかわりに、他の駒をサイコロの目だけ進めることもできます。サイコロの目に一致する駒もなく、その数だけ進める駒もないとき、一番上がりより遠いポイントの駒を上がりにできます。
- ⑨ 1がりの途中でヒットされたとき、その駒が自分のインナーボードに戻ってくるまでは、他の駒を上がりにすることはできません。
- ⑩ 1箇所のポイントには、何回でも駒を入れることができます。15個全部を1箇所に入れてもよろしい。
- ⑪ サイコロの目は、合計して使ってははいけません。
- ⑫ 動かさない目は無効となります。
- ⑬ どちらか一方の目しか使えないとき、選べるときには、大きい目のほうを使わなければならない。
- ⑭ 動かせる目は必ず使います。勝手にパスすることは許されていません。



- ① 両方のサイコロの目を使う限り、どちらを先に使ってもよい。

ダブリング

ゲームの途中、自分がサイコロを振る前に、ダブルを宣言し、賭け点を2倍にすることができず、このとき、ダブルリングキューブの賭け点の面を上にして、バーの相手側に置きます。

このとき、相手はこの賭け点を受け入れゲームを続けるか、賭け点だけの得点を失なってゲームを終了するか2通りの方法があります。

初回のダブルは両者が権利を持っていますが、2回目からは、ダブルリングキューブを持っている側が賭け点をセリ上げることができるだけです。

ダブルリングにより、結果のわかり切ったゲームを早く終了させゲームをスピードアップできるとともに、賭け点をセリ上げていくことにより、スリルのあるゲームとなります。

勝ち方

自分の駒を全部上げたとき、相手の駒の状態により、賭け点の1倍から3倍までの3通りの得点が得られます。

1倍	相手の駒が1個でも上がりになっている。
2倍	相手の駒が1個も上がりになっていないうちに上がる場合で、ギャモンといいます。
3倍	相手の駒が1個でも、自分のインナーボード、またはバーにいるときに上がる場合で、バックギャモンといいます。

以上がルールですが、次のことを覚えておくとゲームがスムーズに進みます。

- 偶数個動かすとき同じ色のポイントへ、奇数個の場合は違う色のポイントへ進みます。
- 6個進めるときは、次の区画内の前と一致する位置に進みます。

MZ-80B

BACK GAMMON

■大藤謙二

オープニング

バック・ギャモンを知ってるかい。そうそうあの西洋双六さ、日本にも昔からあった遊びなのにいつの間にかすたれてしまったものさ。

本物の道具立ては、アタッシュケースの中に三角模様の盤があり、二色のチェッカー（コマ）15個と2個のダイス（サイコロ）、それにダブルング・キューブ1個という簡単なものなんだ。

2人が交互にダイスを振ってやるんだけど、我らマイコン時代人は、やはりマイコンでやりたくなるのが人情、というわけでマイコン版バック・ギャモン・スタート。

プログラムの説明

- ①ダイスはルーレットで何かキーインするとストップする。もし、ダブルングしたいときは **[D]** を押し、相手がダブルングを受けないと云った場合は **[E]** をキーインする。

ゲーム開始



- ②ダイスの目は左右の順に使う。交換する場合は、チェンジ・メッセージに対し **[Y]** を押す。

- ③手のキーインは動かすコマのポインタを入力する。エラーはチェックされ再入力となり、コマが動かないときは **[Z]** を押す。

- ④すべての手をキーインすると、メッセージでこれで良いかどうか聞いてくる。手をやり直したい場合 **[N]** を押す。

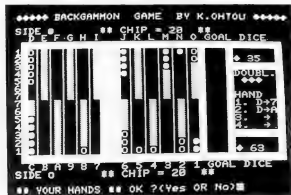
- ⑤ヒットされたコマがリターンできない場合は、リトライのメッセージが出る。やり直しできるときは **[Y]**、どうしてもリターンできないとき **[N]** を押す。

以上、すべてメッセージに従ってキーインすればゲーム進行できます。

エンド・メッセージ

このゲームは1人2役でもできます。腕をみがくにはもってこいでしょう。君は、ランニング派か、バックゲーム派か、それとも、アタッキングかブライムか……ダイレクト・ショット、ヒットアンドプロット、リターン・ヒット、ダブル・ヒット……バックギャモン!!

ゲーム中



MZ-80B バック ギャモン プログラム・リスト

```

10 GOSUB2000
12 DIMMC(2)
14 FORI=1TO2:MC(I)=20:NEXT
20 GOSUB2100
30 GOSUB2200
40 GOSUB2400
42 GOSUB2270
45 GOSUB2500
50 TC=3-TB
52 GOSUB2560
54 IFCH=1THENGOSUB2198:GOTO70
60 GOSUB2300
65 IFAP(TB)>14GOTO80
70 TB=TC
75 GOTO50
80 REM ** CHEER COUNT
85 GOSUB3250
86 GOSUB3190

```

プログラム・リスト

```

90 CURSOR0,23:PRINT"
92 CURSOR0,23:PRINT" ** GAME IS OVER. AGAIN ?(Y OR N)"
94 GETAF:IFA#=""GOTO90
96 IFA#":"N"GOTO20
100 END
2000 REM ** TITLE PRINT
2010 PRINTCHR$(6)
2012 PRINT"***** 0 0 0 0 0 0"
2014 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2016 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2018 PRINT"***** 0 0 0 0 0 0"
2020 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2022 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2024 PRINT"***** 0 0 0 0 0 0"
2026 PRINT:PRINT
2030 PRINT" 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2032 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2034 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2036 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2038 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2040 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2042 PRINT"0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
2044 PRINT:PRINT
2050 PRINT"-----"
2052 PRINT"AUTHOR:CHR$(58)1" LENUT DHIUU "
2054 PRINT" 1981.10.20 "
2056 PRINT"-----"
2060 PRINT" ** KEY IN ANY KEY"
2062 GETAF:IFA#=""GOTO2062
2064 RETURN
2100 REM ** INITIAL SCENE
2102 PRINTCHR$(6):"***** BACKGAMMON GAME BY K.ORTOU *****"
2108 PRINT"SIDE "CHR$(147)1" ** CHIP = "
2110 PRINT" D E F G H I J K L M N O GOAL DICE "
2112 CURSOR0,20:PRINT" C B A 9 8 7 6 5 4 3 2 1 GOAL DICE"
2114 PRINT"SIDE "CHR$(140)1" ** CHIP = "
2116 B#="" :C#=""
2118 FORI=1TO30:B#=#+CHR$(30)+NEXT
2118 CURSOR0,4:PRINT" " :B#
2120 C#=#+CHR$(30)+" " +CHR$(31)+" " " " +CHR$(31)+" " " " +CHR$(31) +CHR$(30) +CHR$(30) +
CHR$(30)
2122 C#=#+" " +CHR$(31)+" " " " +CHR$(31)+" " " " +CHR$(31) +CHR$(30) + " " +CHR$(30) +
" " +CHR$(30)
2124 FORI=1TO7:AS=LEFT$(STR$(I),1)
2126 PRINTA#C#NEXT
2128 C#=#+CHR$(30) +CHR$(31)+" " " " +CHR$(31)+" " " " +CHR$(31)+" " " " +CHR$(30) +CHR$(30) +
CHR$(30)
2130 C#=#+CHR$(31)+" " " +CHR$(31)+" " " " +CHR$(31)+" " " " +CHR$(30) + " " +CHR$(30) +
" " +CHR$(30)
2132 FORI=7TO1STEP-1:AS=LEFT$(STR$(I),1)
2134 PRINTA#C#NEXT
2136 PRINT" " :B#
2138 B#=#+CHR$(30) +CHR$(30) +CHR$(30) +CHR$(30) +CHR$(30) +CHR$(30) +CHR$(30)
2140 CURSOR22,7:PRINTD#
2142 CURSOR22,9:PRINT" DOUBL."
2144 D1=1024:GOSUB3004
2146 CURSOR22,11:PRINT" HAND "
2148 CURSOR22,12:PRINT" 1. "
2150 CURSOR22,13:PRINT" 2. "
2152 CURSOR22,14:PRINT" 3. "
2154 CURSOR22,15:PRINT" 4. "
2156 CURSOR22,16:PRINTD#
2158 GOSUB3190
2160 CURSOR0,23:PRINT" ** KEY IN ANY KEY, THEN GAME START."
2162 GETAF:IFA#=""GOTO2162
2164 RETURN
2180 B#="A0FAF"
2182 MUSTCB#RETURN
2190 B#="A1"
2192 MUSTCB#RETURN
2194 B#="A0+A1"
2196 MUSTCB#RETURN
2198 B#="C7EG+C"
2199 MUSTCB#RETURN
2200 REM ** SET UP INITIAL VALUES
2201 DIMHP(24),SP(24),AF(2),BP(2),PC(24)

```

```

2202 DIMND(24),SD(24),AD(2),BD(2)
2203 DIMNR(24),SR(24),AR(2),BR(2)
2212 DE=1024:D1=DB:GOSUB3004
2214 FORI=1 TO9:PC(I)=I+48:NEXT
2216 FORI=10TO24:PC(I)=I+55:NEXT
2218 FORI=1 TO24:NF(I)=0:SP(I)=0:NEXT
2220 I=1:NF(I)=2:SF(I)=2
2222 I=24:NF(I)=2:SP(I)=1
2224 I=6:NF(I)=5:SP(I)=1
2226 I=19:NF(I)=5:SP(I)=1
2228 I=8:NF(I)=3:SP(I)=1
2230 I=14:NF(I)=3:SP(I)=2
2232 I=12:NF(I)=5:SP(I)=2
2234 I=13:NF(I)=5:SP(I)=1
2236 FORI=1TO2:AP(I)=0:EP(I)=0:NEXT
2238 ID=1
2240 FORI=1TO24:NO(I)=0:NR(I)=0
2241 SD(I)=0:SR(I)=0:NEXT
2242 FORI=1TO2:AD(I)=0:AR(I)=0
2243 BD(I)=0:BR(I)=0:NEXT
2250 I=ID:ID=1
2252 REM ** DOUBLING RIGHT DISP.
2260 U=ID:ID=ID*RETURN
2262 CURSOR32,30-12*ID:PRINT"♦"
2263 TW=7-ID
2264 CURSOR32,30-12*TW:PRINT" "
2265 RETURN
2270 REM ** DOUBLING MARK SET.
2271 CURSOR32,6:PRINT"♦"
2272 CURSOR32,18:PRINT"♦"
2273 RETURN
2299 REM ** MAIN'S ROUTINE
2300 GOSUB3180:GOSUB2380
2304 IFOP=0GOTO2320
2306 CURSOR0,23:PRINT"
2307 CURSOR0,23:PRINT" ** YOUR DICE ** DOUBLING=D:END=E"+CHR$(31)
2308 GOSUB3020
2310 IFA$="E" THENAP(TB)=15*GOTO2372
2312 IFID>26GOTO2314
2313 IFID>18GOTO2320
2314 IFA$="D" GOSUB3000
2320 IFR1<>R2GOSUB3080
2321 OP=1
2322 NN=0:MT=NT:CK=0
2324 NT=NT+1:NN=NN+1
2325 R3=R2:IFNN=1THENR3=R1
2326 GOSUB2385
2327 IFBP(TB)>0GOSUB3200:GOTO2330
2328 GOSUB3160
2329 IFT$="Z" GOTO2340
2330 GOSUB3300
2332 IFCK=1GOSUB2395:GOTO2325
2333 IFCK=2GOTO2360
2334 IFAP(TB)>14GOTO2370
2336 IFNN=NHGOTO2350
2338 GOSUB2385:GOTO2324
2340 GOSUB3400
2342 IFCK=0GOTO2350
2344 GOTO2357
2350 GOSUB3010
2351 CURSOR0,23:PRINT"
2352 CURSOR0,23:PRINT" ** YOUR HANDS ** OK?(Yes OR No)" +CHR$(31)
2354 GETA$:IFA$="" GOTO2350
2356 IFA$(">")N" GOTO2370
2357 NT=MT:GOSUB2390
2358 GOSUB2400:GOSUB2385:GOSUB3180
2359 GOTO2320
2360 CURSOR0,23:PRINT"
2361 GOSUB3010
2362 CURSOR0,23:PRINT" ** RETURN ERROR! RETRY?(Yes OR No)"
2363 GETA$:IFA$="" GOTO2360
2364 IFA$="Y" GOTO2357
2365 CK=0
2370 GOSUB2490:GOSUB2400
2372 RETURN
2379 REM ** COPY BOARD DATA

```

```

2780 FORI=1TO24
2581 NO(I)=NP(I):SO(I)=SP(I):NEX1
2582 FORI=1TO2
2583 AO(I)=AP(I):BO(I)=BP(I):NEX1
2584 RETURN
2585 FORI=1TO24
2586 NR(I)=NP(I):SR(I)=SP(I):NEX1
2587 FORI=1TO2
2588 AR(I)=AP(I):BR(I)=BP(I):NEX1
2589 RETURN
2590 FORI=1TO24
2591 NF(I)=NO(I):SP(I)=SO(I):NEX1
2592 FORI=1TO2
2593 AP(I)=AO(I):BP(I)=BO(I):NEX1
2594 RETURN
2595 FORI=1TO24
2596 NF(I)=NR(I):SP(I)=SR(I):NEX1
2597 FORI=1TO2
2598 AR(I)=AR(I):BP(I)=BR(I):NEX1
2599 RETURN
2600 REM ** CREAT SCENE
2602 OUT232,128
2604 KK=2
2605 FORI=1TO24:KK=3+K
2607 IFSP(I)< SR(I)GOTO2410
2608 IFNP(I)< NR(I)GOTO2410
2609 GOTO2415
2610 A=147:IFSP(I)=1THENA=148
2611 FORJ=1TO7:IFJ:NP(I)THENA=33+J
2612 GOSUB100
2614 POKEI,J,A:NEXTJ
2616 NEXTI
2630 FORI=1TO2
2632 IFAP(I)=AR(I)GOTO2444
2634 I=0:A=149-I:FORJ=1TO15:J=J
2636 IFI:AP(I)THENA=32
2638 I=I-5 THENI=2:J=J-5
2639 IFI:10THENI=2:J=J-10
2640 I=545I-520+I+1
2641 J=1-40+(J-1):GOTO2445
2642 I=11+40*(J-1)
2643 POKEI,J,A:NEXTJ
2644 NEXTI
2650 FORI=1TO2
2652 IFBP(I)=BR(I)GOTO2464
2654 A=149-I:FORJ=1TO7
2656 IFI:BP(I)THENA=30
2658 J=53742+40*(J-1)
2659 I=53782-40*I
2660 IFI:1+40*(J-1):GOTO2465
2661 J=I-40*(J-1)
2662 I=I-40*(J-1)
2663 POKEI,J,A:NEXTJ
2664 NEXTI
2670 OUT232,0:RETURN
2679 REM ** DISPLAY HAND
2680 I1=1+N
2682 I1=1+N
2683 IFBV<1THENP1=134:GOTO2486
2684 IFBV<1THENP1=134:GOTO2486
2685 P1=PC(BV)
2686 A$=CHR$(P1)
2687 CURSOR35,11:PRINTA$+" "+A$
2688 RETURN
2690 FORI=1TO24:NR(I)=NO(I):SR(I)=SO(I)
2692 NEXT
2693 FORI=1TO24:AR(I)=AO(I):BR(I)=BO(I)
2694 NEXT
2696 RETURN
2698 REM ** INITIAL RULE1
2702 CURSOR0,23:PRINT" ** GAME OPENING
2704 TB=1:P1=INT(6*RND(1)+1)
2705 C1$=LEFT$(STR$(P1),1)
2706 CURSOR4,18:PRINTC1$
2708 NR(I)=IFAS$=""GOTO2504

```

```

2510 TB=2:R2=INT(6*RND(1)+1)
2511 C2%=LEFT$(STR$(R2),1)
2512 CURSOR34,6:PRINTC2%
2514 GETA$:IFA$=""GOTO2510
2516 TB=1:IFR1>R2GOTO2534
2518 TB=2:IFR2>R1GOTO2528
2520 TB=0:GOSUB3000
2522 CURSOR0,23:PRINT** AGAIN OPENING DICE!
2524 GOTO2504
2526 R%=R1:R1=R2:R2=R3
2528 C1%=C1$:C1%=C2$:C2%=C3%
2530 CURSOR34,30-12*TB:PRINTC1%:C2%
2532 DB=0:NH=2
2534 RETURN
2536 REM ** PRIME CHECK
2538 DB=0
2540 IF DB<10 THEN RETURN
2542 I1=19:I2=24
2544 IF TB=2 THEN I1=1:I2=6
2546 FOR I=1 TO I2
2548 IF SP(1)/I<YCGOTO2578
2550 IF NP(1)/I<YGOTO2578
2552 NEXT I
2554 DB=1
2556 RETURN
2558 REM ** DOUBLING SE
2560 IF DB=1023 THEN DB=2:GOTO3004
2562 DB=DB+DB
2564 DB%=INT(DB+0.991)
2566 DB%=RIGHT$(DB%,4)
2568 IF DB=1023 THEN DB=" ***"
2570 CURSOR32,9:PRINTCDB%:DB=DB
2572 DB=3-TB:GOSUB2260
2574 RETURN
2576 CURSOR33,30-12*TB:PRINTCHR$(C1)
2578 FOR I=1 TO 10: NEXT I
2580 CURSOR33,30-12*TB:PRINTCHR$(C2)
2582 RETURN
2584 REM ** DICE RULLET AT MACHINE SIDE
2586 M3=30-12*TB
2588 R1=INT(6*RND(1)+1)
2590 C1%=LEFT$(STR$(R1),1)
2592 CURSOR34,M3:PRINTC1%:C1%
2594 GETA$:IFA$=""GOTO3022
2596 M1=INT(50*RND(1)+19)
2598 FORM=1 TOM1:R2=INT(6*RND(1)+1)
2600 C2%=LEFT$(STR$(R2),1)
2602 CURSOR34,M3:PRINTC1%:C2%
2604 NEXT
2606 NH=2:IFR1=R2 THEN NH=4
2608 RETURN
2610 REM ** DICE SELECTION
2612 GOSUB3010
2614 CURSOR0,23:PRINT"
2616 CURSOR0,23:PRINT** CHANGE DICE (Yes OR No)?":CHR$(31)
2618 GETA$:IFA$=""GOTO3080
2620 IFA$="Y" GOTO3092
2622 RETURN
2624 R%=R1:R1=R2:R2=R3
2626 C1%=C1$:C1%=C2$:C2%=C3%
2628 CURSOR34,30-12*TB:PRINTC1%:C2%
2630 RETURN
2632 REM ** ADDRESS (I,J) CALCULATION
2634 IF I<7 GOTO3114
2636 IF I<13 GOTO3112
2638 IF I<19 GOTO3110
2640 IJ=40*IJ+2*I+53386:GOTO3116
2642 IJ=40*IJ+2*I+53384:GOTO3116
2644 IJ=54034-40*IJ-2*I:GOTO3116
2646 IJ=54036-40*IJ-2*I
2648 RETURN
2650 REM ** KEY IN MAN'S HAND
2652 GOSUB3010
2654 CURSOR0,23:PRINT"
2656 CURSOR0,23:PRINT** KEY IN YOUR HAND ":CHR$(31)
2658 GETT$:IFT$=""GOTO3160

```

プログラム・リスト

```

3164 IFT4="Z" GOTO3178
3165 AV=ASC(1$)
3166 IF AV<49 GOTO3174
3167 IF AV>79 GOTO3174
3168 IF AV=58 THEN AV=AV-48: GOTO3178
3169 IF AV=64 THEN AV=AV-55: GOTO3178
3174 GOSUB2180: GOTO3160
3178 RETURN
3179 REM ** CLEAR HAND DISPLAY
3180 OUT232,128
3184 FOR I=1 TO 4: I1=53763+40*(I-1): I2=11+2
3186 POKE I1,32: POKE I2,32: NEXT
3188 OUT232,0: RETURN
3190 REM ** DISPLAY CHIP NUMBER
3191 M1=MC(2)
3192 C3$=" "+STR$(M1): C3%=RIGHT$(C3$,4)
3194 CURSOR2,2: PRINT C3$
3195 M2=MC(1)
3196 C3$=" "+STR$(M2): C3%=RIGHT$(C3$,4)
3198 CURSOR2,21: PRINT C3$
3199 RETURN
3200 BV=R3: AV=R3
3202 IFTB=2 THEN GOTO3208
3204 BV=25-R3: AV=BV: T$="★"
3208 RETURN
3210 CH=0: IFTB=260 TO 3212
3211 I1=AV+1: I2=6: GOTO3213
3212 I1=19: I2=AV-1
3213 FOR I=1 TO 12
3214 IF SP(I)=TB THEN CH=1: GOTO3218
3215 NEXT
3218 RETURN
3250 REM ** COUNT CHIPS
3251 CN=1: TC=3-18
3252 IF A$="E" GOTO3268
3255 IF AP(TC): GOTO3268
3256 IF TB=260 TO 3258
3257 I1=1: I2=6: GOTO3260
3258 I1=19: I2=24
3260 FOR I=1 TO 12
3261 IF SP(I)=TC GOTO3267
3262 NEXT
3264 IF BP(TC): GOTO3267
3266 CN=2: GOTO3268
3267 CH=1
3268 IF DB=100 THEN DB=1
3270 CC=INT(CN*DB+0.1)
3271 MC(18)=MC(18)+CC
3272 MC(TC)=MC(TC)-CC
3273 CURSOR0,11
3275 ON CN GOTO3276,3277,3278
3276 PRINT" ** CONGRATURATION! " : GOTO3280
3277 PRINT" ** CONGRATURATION! GAMMON WINNING! " : GOTO3280
3278 PRINT" ** CONGRATURATION! BACK GAMMON WINNING! "
3280 PRINT" ** YOU GET " ; CC ; " CHIPS"
3281 RETURN
3290 REM ** CHECK MOVING
3300 CH=0: IF BP(TB): GOTO3370
3302 IF SP(AV): GOTO3390
3308 IF BP(AV): GOTO3390
3309 IF TB=1 THEN BV=AV-R3: IF BV<1 GOTO3320
3310 IF TB=2 THEN BV=AV+R3: IF BV>24 GOTO3330
3311 IF SP(BV): GOTO3340
3314 IF SP(BV): GOTO3340
3315 IF BP(BV): GOTO3350
3316 GOTO3390
3320 FOR I=1 TO 24: IF SP(I)=TB GOTO3390
3321 NEXT
3324 IF BP(TB): GOTO3390
3325 IF BV<0 GOTO3360
3326 IF BV=25 GOTO3360
3327 GOSUB2180: IF CH=0 GOTO3360
3328 GOTO3390
3329 FOR I=1 TO 18: IF SP(I)=TB GOTO3390
3331 NEXT
3332 GOTO3324

```

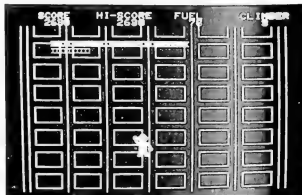
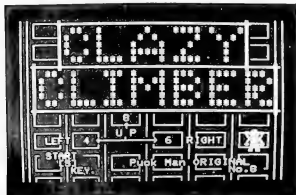
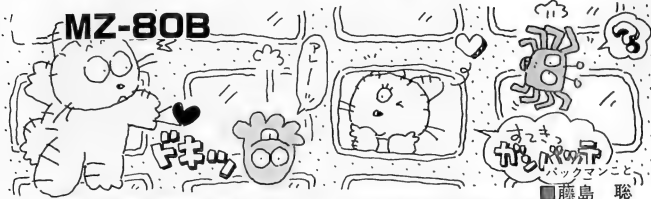
```

3340 GOSUB2190:NP(BV)=NP(BV)+1:SP(BV)=TB
3342 NP(AV)=NP(AV)-1:IFNP(AV)=0THENSP(AV)=0
3344 GOSUB2480
3348 GOTO3395
3350 GOSUB2194
3351 NP(BV)=1:SP(BV)=1B:TC=3-TB:BP(TC)=BP(TC)+1
3352 GOTO3342
3354 GOSUB2190
3356 AP(TB)=AP(TB)+1
3358 GOTO3342
3360 BV=AV:1%="4"
3361 IF SP(BV)=TBGOTO3375
3362 IF SP(BV)=0GOTO3375
3363 IF NP(BV)=1GOTO3377
3364 CK=2:GOSUB2180:GOTO3395
3365 GOSUB2190:NP(BV)=NP(BV)+1
3366 SP(BV)=TB:GOTO3379
3367 GOSUB2194:NP(BV)=1:SP(BV)=TB
3368 TC=3-TB:BP(TC)=BP(TC)+1
3369 BP(TB)=BP(TB)-1:CK=0
3370 GOSUB2480:GOTO3395
3372 GOSUB2180:CK=1
3374 RETURN
3400 REM ** ALTERNATIVE ?
3410 CK=0
3420 RETURN

```

CRAZY CLIMBER

クレイジークライマー



ビルを登ることが何の得になるのでしょうか……。お金をもらえるわけでもなく、人々の役に立つわけでもない……、ただ自分の命を冷や冷やさせ、そして落としていくだけです……。

しかし、あなたは登ります。意味がなくてもとにかく登ります。きっと、あなたはまきがいなのでしょう。それではあなたもキチガイになってビルに登ってください。

1. ゲームの説明

11月号で誰かが言っていました。「合計16個のキーを動かせるでしょうか」と。しかし16個も必要ありません。無理にそのままだにすることは無いのです。このクレイジークライマーでは、なんと片手で塩せんべいを食べながらでもできるよう、なんとキーはたったの3個にしてみました。人間は、テン・キーの4と6で左右に動き、8で上へ登るのです。すこし、これでは簡単すぎるではないか、ですって？、はい、え、それならどうしてあります。

ゲームセンターにでもこれと同じ名のゲームはありますが、同じような態勢で同じように使えるというだけで、かなり違います。まず窓が閉まりません。落ちてくる物しかありません（本物でいうとしらけコンドル、おじいちゃんなどが出てこないのです）。

しかも何であんなに落ちてしまいます。床までいくと、バックを捕まえても逃がなければしごとをとり戻されます。人間の足が動きません。登り方も違います。

さて、落ちてくる物ですが植木ばち、鉄釘その他いろいろあります。特にインペーダー、ギャラクシアン、バックマンとオバケ、クレイジークライマー（自殺する）と特別出演のゲストも取りそろえております。

RUNさせてしばらくすると画面が出てきます。説明を読んだら、[5]キーでスタートです。快い(?)音楽と共にゲームがスタートします。さあ、ここからはあなたの世界です!

窓と窓に手をはさんでは使えず、両手が窓にしっかりかかった状態でこそ登れます。というのは本物が手づかまって登るのに対し、このクライマーは吸盤を手足にくっつけており、それで登ります。よって壁にはひっつかず、窓だけつきます。

エネルギーがあり、1階(?)登ると2増えます。登らなるとどんどん減って、0になると落ちてしまいます。

クライマーは3人いて、10,000点で1人増えます。

障害物は全部で10種類、10面目まで違うパターンと音楽で楽しめます。10面目を越すと、ずーっと10面目が続きます。

戦い終わって、ハイ・スコア5位以内に入っていると名前がかけます。テン・キー[4]と[6]でカーソルを左右に動かし、[SPACE]で記入してください。BGMがいいでしょう。

2. プログラム

BASICでメインが書かれていて、登るとき、落ちるとき

表1

0	REM 初心者の人に聞きます。ここでは何を すると思いますか?
10~ 30	第1初期設定
40~ 90	第2 "
95~ 98	ゲーム・スタート
100~ 215	メイン
220~ 750	障害物
800	屋上処理
1000~1950	障害物移動
2000~2940	登る処理
4000~4250	スタート・デモ
6000~6060	屋上についた処理
6100~6980	画面クリア, MUSIC処理
7000~7020	死んだとき
7030~7085	ゲーム・オーバー
7090~7490	名前を書く
7500~7540	まだ生きているとき
8000~8060	ビルのスタート (地上) の枠
9000~9400	データ書き込み (設定)
9500~	データ

の音、落ちる時のグラフィックRAMのスクロール、
登る時のビルのスクロールのみマシン語です。キーボ
ードからポチポチ入れたため、フローチャートはちょっとか
んべんしてください。その代わりというのはなんですが、
どこでどんな処理をしているか、表1に表します。

REM女は素通りするようになっているので入力しなく
ても大丈夫です。

また、グラフィックRAMは移さず、ビルをVRAM (キ
ャラクタ) に書いてマシン語でスクロールしているため、
他機種への移植は難しいどころか、できないかもしれま
せん。

3. 必勝法

●植木ばち

とにかく登って、来たらハット横に移る。

●鉄鋼(小)

植木ばちとほとんど同じで、真上に落ちてきたら短い方
に逃げる。

●仲良しインペダー

5~7階登り、ハット隣りに移る。これを繰り返して登
る。

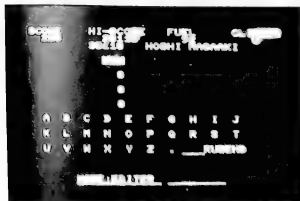


表2

1面	6100~6140	} 本物と同じ
2面	6200~6230	
3面	6300~6350	} 本物と似ている
4面	6400~6480	
5面	6500~6540	} 少女人形
6面	6600~6630	
7面	6700~6740	} ムーンライトキッス
8面	6800~6820	
9面	6900~6920	} 哀戦士
10面	6950~6980	

●自殺人間

2階登り、ハット隣りに移る。少しでもミスるとすぐひ
っかかったり、エネルギーがなくなるので真剣にやる。

●バックマン&オバケ

植木ばちと同じ。しかしはさむように落ちてきたときは
バックマンとオバケの間で落ち切るのを待つ。

●バーベル

鉄鋼と同じ。

●イカリ

あわてず、よく位置を確認し、避けるなら必ず右に避ける。

●ギャラクシアン

ギャラクシアンと逆の方向に逃げ、捕まらないと自信の
あるところで、ギャラクシアンが落ち切るまで登る。

●Y-Hミサイル

第一分節、つまり1回目で止まるところで右か左に1つ
避ける。失敗すると落ちるので気をつける。また、これは
頭に当たらないと落ちないことを頭にいれてやるとよい。

●鉄鋼(大)

落ちる場所が決まるまでまん中で待ち、落ちる場所がわ
かったらすばやく避けて、登る。

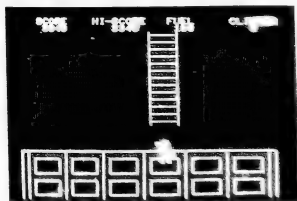
4. プログラムの改造

●40のMCの値を増やせば人間の数が増える。

●300~750の行番号の組み合わせを変えると障害の出る順
が変わる。

●6000番台の音楽データを好きな曲にしよう。

●9000番台のデータをいじって、メチャクチャの絵にしな
りして遊ばしよう(アホか)。特にマシン語データを変
えると異状(暴走など)が起こってなお面白いので、後
者をやるのはアホかキチガイでしょう。



GRAZY GLIMBER

5. その他

画面クリアの音楽を参考までに表2に表します。適当に変えてみてください。BASICを入れて、すぐ実行すれば以外は、GRAPH 11を実行してください。これを実行すればどうなるかわからない人は、GRAPH 12を実行してからRUNしてみてください。**BREAK**でプログラムを止めたら、超高速オート・リビートがかかっているのを、**F7**を押して元に戻してください。オート・リビートをするには、\$0952から\$0953を0にすればできます。また、それぞれの元の値はそれぞれ166、115です。

6. おわりに

そうとう手つきがしてあるようにみえますが、本当は手抜きなんかしています。しかし、結構面白いのですねー。自分でも熱中してしまい、5人にして、やっと193,400点いきました。友人に見せたら、「あ、バックマシが出た、インペータが出た、人か何人も自殺してる、下にはどんなものがたまるとるんや」とか、いろいろ楽しんでいました。僕はBASIC振ですが、今頃はできたマシン語で作りたいなーと思えます。

P.S. プログラム・リストをプリントでとらせてくれた
URBAN 電子の皆様、アイデアを提供してくれた平嶋義博君、
遊んだ感想を聞かせてくれた中島聡君、小松慎一君、五反
田裕君に誌上を借りてお礼申しあげます。

●參考資料

- 1) ゲーム：バックマン、ラリーX、ギャラクシアン、
クレイジークライマー、スペース・インベーダ、ギャラガ
2) 歌謡曲：少女人形、哀戦士、ムーンライトキス
3) シューブ：アプリケーション・プログラム「MUSIC」

クレイジークライマー プログラム・リスト

[illegible]

プログラム・リスト

283

プログラム・リスト

84

プログラム・リスト

285

CRAZY GLIMBER

プログラム・リスト

```

9200 US="" :FOR I=1 TO 40:READY:US=US+CHR$(Y):NEXT:US=US+US
9210 V$="" :FOR I=1 TO 52:READY:V$=V$+CHR$(Y):NEXT
9220 N1$="" :FOR I=1 TO 52:READY:N1$=N1$+CHR$(Y):NEXT
9230 N2$="" :FOR I=1 TO 52:READY:N2$=N2$+CHR$(Y):NEXT
9240 O$="" :FOR I=1 TO 24:READY:O$=O$+CHR$(Y):NEXT
9250 P$="" :FOR I=1 TO 22:READY:P$=P$+CHR$(Y):NEXT
9260 Q$="" :FOR I=1 TO 48:READY:Q$=Q$+CHR$(Y):NEXT
9270 R$="" :FOR I=1 TO 16:READY:R$=R$+CHR$(Y):NEXT
9280 S$=US+V$
9290 FOR I=1 TO 30:READY:Z:CX(1)=Y:CY(1)=Z:NEXT
9400 RETURN
9500 DATA 219,232,203,255,211,232,33,191,211,17,231,211,1,112,3,237
9510 DATA 184,35,84,93,19,1,39,0,175,119,237,176,201
9520 DATA 219,232,203,255,211,232,33,159,254,17,63,255,1,0,20,237
9530 DATA 184,35,84,93,19,1,159,0,175,119,237,176,201
9600 DATA 0,0,56,125,125,57,24,29,14,7,3,0,0,0,0,0
9610 DATA 0,0,0,1,1,1,3,3,60,126,255,255,255,255,126
9620 DATA 126,255,255,255,255,255,255,255,231,231,231,231,195,195,195,195
9630 DATA 0,0,0,126,126,126,0,0,0,192,224,112,56,24,26,62
9640 DATA 62,28,0,126,126,126,126,192,192
9650 DATA 0,0,0,1,1,1,0,0,0,3,7,14,28,24,56,124
9660 DATA 124,56,0,1,1,1,3,3,60,126,255,255,255,255,126
9670 DATA 126,255,255,255,255,255,255,255,231,231,231,231,195,195,195,195
9680 DATA 0,0,28,190,190,190,156,24,56,112,224,192,0,0,0,0,0
9690 DATA 0,0,0,126,126,126,126,192,192
9700 DATA 7,12,3,19,9,37,19,63,32,63,16,16,8,8,15
9710 DATA 0,192,192,0,16,36,72,144,252,4,252,8,8,16,16,240
9720 DATA 0,255,115,33,33,115,255,0,0,255,156,8,8,156,255,0
9730 DATA 0,255,231,66,66,231,255,0,0,255,57,16,16,57,255,0
9740 DATA 0,255,206,172,172,206,255,0,24,60,126,219,255,36,90,165
9750 DATA 16,9,71,54,255,191,160,25,128,3,135,198,247,211,86,140
9760 DATA 240,252,254,246,254,156,102,3
9780 DATA 0,15,31,63,127,255,255,254,254,254,252,252,124,56,24,6
9790 DATA 0,224,240,246,252,254,254,254,254,254,126,126,124,56,48,32
10000 DATA 7,31,63,115,226,226,227,243,255,255,255,255,255,125,49
10010 DATA 24,248,252,206,175,135,183,207,255,255,255,255,222,140
10020 DATA 56,116,250,255,254,124,56,0,0,0,0,255,0,0,0,0
10030 DATA 28,58,125,255,127,62,28,0,3,4,4,3,1,9,15,1
10040 DATA 129,129,193,225,127,63,15,1,128,64,64,128,0,0,224,32
10050 DATA 2,6,14,252,248,224,0
10055 DATA 0,0,0,97,97,127,127,1,1,31,32,64,65,66,124
10060 DATA 102,102,102,102,255,255,153,153,255,255,255,126,255,126,60,24
10070 DATA 0,0,0,0,134,134,254,254,128,128,248,4,2,130,66,62
10080 DATA 60,66,129,129,129,129,66,60,24,24,24,24,153,90,60,24
12000 DATA 4,13,7,13,10,13,13,13,16,13,19,13,22,13,25,13,28,13,31,13
12010 DATA 4,15,7,15,10,15,13,15,16,15,19,15,22,15,25,15,28,15,31,15
12020 DATA 4,17,7,17,10,17,13,17,16,17,19,17,22,17,25,17,28,17,31,17

```

MZ-80B

DONKEY KONG

モドキ

プロローグ

任天堂星での激しい戦いが終わり、3ヶ月がやみくもに過ぎ去った。兄、ドンキーは、弟、ドンキーモドキに一通の遺書を残して死んでいった。遺書の内容はこうであった。「弟よ、私の意志を受け継いで、コングモドキを倒してくれ…」。

弟は旅立った。新しく始まるMZ星での戦いのために…。

■居候VALZA

〔状況〕

前回の戦いとほぼ同様である。もし、前回の戦いを知らないような人がいるとしたら、その人はいますぐゲーム・コーナーへ行って再現されているゲームを見に行ったほうがよい。

〔解説〕

★コングモドキとは

任天堂星のムースニー帯に住む平面ゴリラ科、原産地はサザンブトン島である。

●体長…3 m 40cm

●体重…0 kg

繁殖力が非常に弱くめったに人に姿を現さない。二次元の世界の動物である。超能力を持ち、樽や火などをいたる所に空間転送することができる。それ以上のことはまだ何も説明されていない。

ドンキーコングは有名なので（I/Oフラザにもときどき出ている）、知らない人は100円を持ってゲーム・コーナーへ行って見てください。

このゲームはドンキーコングになろうとして30秒間苦しみましたが、どうしてもなりきれず、結局「ドンキーコングモドキ」ということになりました。本物と違う点は、

①ジャンプして鉄骨に当たると死ぬ（人間の世界でこのようなことをしたら無事ではいられません）。

②ジャンプして階段に飛び乗ることができる（世間では常識）。

③ジャンプして階段から飛び降りることができる（世間では常識）。

④飛び降りて加速がつくと地面に激突する前に死ぬ（結果は同じなのでかんべんしてください）。

⑤[5]のキーを押して続けると連続してジャンプし続けることができる（3回以上連続してジャンプをすると死にます…バグではありません）。

⑥鉄骨が1個分あいているところはまたいで行けます（ただし、途中でキーから指を離すと死にます）。

⑦鉄骨が2個分あいているところは [5] と [4] または [5]

1 ゲームの説明

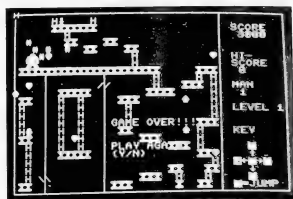


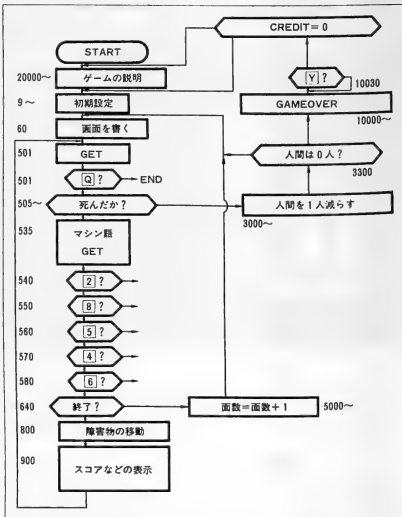
表 1

主な変数名	内 容
S	スコア
HI	ハイ・スコア
HX(h)	はしごの一番上のところの座標
HY(h)	
MA\$	人間のキャラクタ
HO\$	
EA\$	デコレーション用
FE\$	
KO\$	
GI\$	
C\$	クリア用(空白)
MA	人間の数
D	滞空時間
N	面数
LV	LEVEL
CR	CREDITの数
E	乱数
KU	くいの数
EL	エレベータの位置
X	
Y	いま、人間のいる座標

マシン語ルーチン

アドレス	内容
B000~BFFF	画面DATA
C000~C064	楯の移動
C065~C07E	<input type="checkbox"/> と <input type="checkbox"/> の交換
C090~C0E3	キースキャン
C100~C164	火の移動
C170~C1CE	画面転送
C300~C460	エキスパンダー移動

フローチャート



と **6** を同時に押し続けければ飛べます（“飛ぶ”と言うよりは、“ひっかかってからはい上がる”という感じになります
が…）。

③横には鉄骨があるが下にはない状態では、鉄骨のある方へ歩いて行く（**4** か **6** のキーを押す）ことをすれば、はい上がれます。

⑨ 火が上がって行くことは決していない（現実の世界では常識）。

⑩最大かつ重大なる欠点、ハンマーがない（あんな重たい物を持てるか！）。

①エキスパンダーはランダムに出る（その方が面白いんだ！）。

⑫レディの落とし物は1種類しかない(あんなもの何種類あっても同じだ!).

⑩エレベーターは1つの上下するところに1つしか存在しない。(代わりにスピードは速くしてあります)

④4 面目のくいが16本ある（まあ、いいじゃないですか）。
⑤コングに当たっても死を招くことはない（人間はコング

より強いのだ!)。

④全体的に画面がくずれている (タテが25行ではやりづら

①画面左上に **[H]** が出る(ゲームの実行にあたってじゃま

なものではないので取りませんでした)。どうしても取りたいと言う人はBASICを解説してみてください。BASICの

勉強になります。

⑩BONUSがない（人生は長い のんびりと生きよう）。

⑩MUSICが本物とは似ても似つかない(音程が狂っている程度ではないのです。曲そのものが違っているのです)。

⑩ハイ・スコアで名前を書くことができない(ハイ・スコアさえ残ればいいじゃないですか)。

④人間の形が変わらない (ウッ…ここだけじゃないか、文句がつけられないのは…).

以上、長々と書きましたが、このようにしてみると、このプログラムは世間の条理に合っていると言えるでしょう（こんなことでも言わないとこのプログラムがかわいそうです）。

2 プログラム

できる限りグラフィックRAM 2を使わないように作ったので、「グラフィックRAM 2のある人はより美しく、そうでない人はそれなりに、できます」

ない人はそのまま、または以下のように変えてください。

```
498 IF N=3 THEN GRAPH 11,01:LINE 40,72,  
40,195:LINE 14*8,72,14*8,195
```

55000 GRAPH I1.01

図 1

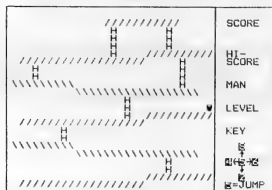


図 2

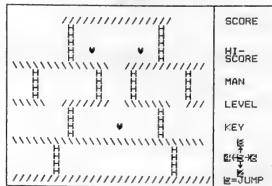


図 3

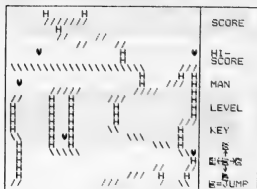
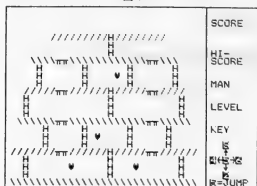


図 4



それから、LIST60000以降は入力なくても結構です。REM文ですが、文中にREMERKと言う物があります。しかし、このような命令はなく、REM+ERKなので勘違いしないでください。入れるときはREMでもREMERKでもREMEDYでもREMEMBERでもREMITでもREMOVEでもREMOTEでも最初にREMさえつけば何でもかまいません。

ゲームのスタートは **[O]** で、CREDITは **[+]** で増えます。

BREAKですが、**[SHIFT] + [BREAK]** だと、ときどきSB-1520に落とし入れられるので、**[Q]** を押せば止まるようにしてあります。再開したいときはダイレクト・モードで、GOTO60 **[CR]** をしてください。続きの面の最初から始まります。

3 プログラムの入力の仕方

- 1) C000～C460までのマシン語を入力します。
- 2) 画面DATAを作りますが、作り方は、LIST1を入れRUNさせると、「ナンメン」と聞いてくるので、1面目の場合**[1]** **[CR]** として画面上に図1を書き込み、**[CR]** を押せばできあがりです。同じように図2～4も入れてください。
- 3) ここまでができれば、マシン語をSAVEします。S-Adr. \$B000, E-Adr. \$C460, J-Adr. \$1300でSAVEしてください。
- 4) BASICプログラムを入れます (グラフィックRAM 2がない人は前述したようにしてください)。
- 5) BASICをSAVEします。

リスト 1

```

10 INPUT"ナンメン":A
20 U=4096*11+256*4*(A-1):O=53248-U:OUTO
232,128:FOR I=U TO U+1000:POKE I+O,
PEEK(I):NEXT:OUTO232,O
30 INPUT":A#
40 OUTO232,128:FOR I=53248 TO 54248:A=
PEEK(I):IF A=0 THEN A=32
50 POKE I-O,A:NEXT:OUTO232,O:PRINT CHR#
(6):GOTO 10
    
```

4

必勝法

階段の下から2段目の所において、下で棒や火が通り過ぎるのを待つという方法があります。

それから、1面目では300点、2面目では3,000点、3面目では1,100点、4面目では1,600点という高得点を出す方法がありますが、ここでは述べずにおきます。ヒントとしては、キーに手を触れなくてもできること、人間は常に一定の方向に動いているのです (キーに手を触れず動ける方法は限られているので、すぐにわかるでしょう)。

それから、2面目では右側の表示の **[4]** の上に乗ることができます (このことができるのはまれで、友人が言うには20%の確率でできると言っています。ちなみに人間は3人しかいませんが...)。このときにBONUSがほしい人は、

```

950 IF CHARACTER$(X,Y+1)="E" THEN S=S+
5000:GOTO 5000
    
```

を追加してください。

6 プログラムの改造

表1とプログラムを見ればできると思います。両面の改造は、両面DATAと行番号14~17のDATA文と3面はC300からC460のエクスパンダー移動ルーチンを変えてください。

6 おわりに

最初は1面目だけを作って終わろうとしていたものをついに4面目まで作るはめになってしまいました。そのせいもあって、かなりのマカロニ・スパゲッティ・プログラムなので、どこに何があるのか作者でもわからないほどですが、頑張ってフローチャートなどを書いておきます。ちなみに私のHI-SCOREは、止ま堂々とやって9,600点、必勝

法を駆使して19,200点です。

最後に、このような中途半端なプログラムでなく、完全とまでは、いませんが、完全に近いドンキーコングの出現を願います(結局、自分も他人をあてにしている)。

7 お礼の言葉

このプログラムの作成にあたりMZ-80Bシステムを快く貸して下さったVALZAの目崎健市氏ほか、協力をしてくださった皆様に謝意を借りてお礼を申し上げます。

参考文献

DI/O, '81年4月号~'82年1月号

残りはLIST 61000以降に掲載

DONKEY KONG モドキ BASIC プログラム・リスト

```

1 LIMIT$=AFF
2 GRAPH12,012,C:GRAPH11,012,C
3 CONSOLEC40
9 HI=0:DIHMX(100,5),HY(100,5):EL=0
10 RESTORE:I=0:U=1
11 READH:READV:IF(H=100)+(V=100)THEN U=U+1:I=0:GOTO11
12 IF(H=200)+(V=200)THENZ0
13 HX(I,U)=H:HY(I,U)=V:I=I+1:GOTO11
14 DATA23,19,8,15,17,11,4,7,25,6,15,2,23,2,100,100
15 DATA7,18,24,18,9,13,22,13,4,8,13,8,18,8,26,8,9,2,22,2,100,100
16 DATA1,12,2,17,12,17,12,28,21,29,8,27,8,100,100
17 DATA5,7,12,7,19,7,26,7,3,11,27,11,6,15,12,15,19,15,25,15,3,19,27,19,100,100
18 DATA200,200
20 HD$=CHR$(#08)+CHR$(#14)+CHR$(#2A)+CHR$(#2C)+CHR$(#5A)+CHR$(#34)+CHR$(#6E)+CHR$(#D5)+CHR$(#AB)+CHR$(#FF)
21 HD$=HD$+CHR$(#FF)+CHR$(#E7)+CHR$(#FF)+CHR$(#FF)+CHR$(#00)
23 EA$=STRING$(CHR$(0),8)+STRING$(CHR$(255),8)
24 FE$=CHR$(#FF)+CHR$(#FF)+CHR$(#24)+CHR$(#18)+CHR$(#18)+CHR$(#24)+""*nn
20 REMARK
21 C$=STRING$(CHR$(0),8)
22 MA$=CHR$(#1C)+CHR$(#3E)+CHR$(#1C)+CHR$(#08)+CHR$(#3E)+CHR$(#08)+CHR$(#14)+CHR$(#22)
23 GI$=CHR$(#1C)+CHR$(#1C)+CHR$(#08)+CHR$(#1C)+CHR$(#2A)+CHR$(#08)+CHR$(#14)+CHR$(#22)
25 KO$=CHR$(#01)+CHR$(#03)+CHR$(#02)+CHR$(#1F)+CHR$(#3F)+CHR$(#3F)+CHR$(#77)+CHR$(#67)
26 FO$=FO$+CHR$(#67)+CHR$(#67)+CHR$(#67)+CHR$(#0F)+CHR$(#1F)+CHR$(#3E)+CHR$(#3C)+CHR$(#7C)
27 FO$=FO$+CHR$(#C0)+CHR$(#F0)+CHR$(#A0)+CHR$(#FC)+CHR$(#FE)+CHR$(#FE)+CHR$(#F7)+CHR$(#F3)
28 FO$=FO$+CHR$(#F3)+CHR$(#73)+CHR$(#F3)+CHR$(#F8)+CHR$(#FC)+CHR$(#3E)+CHR$(#1E)+CHR$(#1F)
29 GOSUB20000
30 S=0:MA=3:D=0:N=1:LV=1
31 QNNGOSUB30000,30100,30200,30300,65535
360 REM
370 C$=STRING$(CHR$(0),8)
380 MA$=CHR$(#1C)+CHR$(#3E)+CHR$(#1C)+CHR$(#08)+CHR$(#3E)+CHR$(#08)+CHR$(#14)+CHR$(#22)
497 GOSUB55000
498 IFN=3THENGRAH12,012:LINE40,72,40,195:LINE14*8,72,14*8,195:GRAPH11,012
499 MUSIC"D3A1":X=2:Y=21:D=0:KU=16
500 REM
501 GETA$:IFA$="0" THENPRINTCHR$(5):END
505 A$=CHARACTER$(X,Y)
510 IF(A$="/" )+(A$="\" )+(A$="H" )+(A$=" " )+(A$="V" ) THENGOTO520
515 GOTO30000
520 IF(A$="/" )+(A$="\" ) GOTO527
525 GOTO530
527 IFCHARACTER$(X,Y+1)<>"H" GOTO30000
530 GOSUB1200
535 USR(#C090)
540 IFPEEK(49664)=1GOSUB1000
550 IFPEEK(49668)=1GOSUB1040
    
```

```

560 IFPEEK(49666)=1GOSUB1070
570 IFPEEK(49665)=1GOSUB1110
580 IFPEEK(49667)=1GOSUB1150
590 IFCHARACTER$(X,Y+1)<>" " THENEND=0:GOTO610
600 D=D+1:IFD=3:GOTO3000
610 IFCHARACTER$(X,Y+1)="●"+(CHARACTER$(X,Y+1)="▲")THEN GOSUB1230
640 IFY=1GOTO3000
650 IFCHARACTER$(X,Y)="♥" THENS=S+500:MUSIC"A1" :CURSORX,Y:PRINT " ";
660 IFCHARACTER$(X,Y+1)="—" THENGOTO3000
670 CURSOR0,0:PRINT " ";
800 DNEGOSUB40000,40100,40200,40300
900 CURSOR32,3:PRINTS;CURSOR32,8:PRINTH1;CURSOR32,13:PRINT"LEVEL";LV:CURSOR32
11:PRINT6;
997 GOTO500
1000 IF(CHARACTER$(X,Y+1)="H")+(CHARACTER$(X,Y+2)="H") GOTO1020
1010 RETURN
1020 POSITIONX#B,Y#B:PATTERN-B,C#;Y=Y+1:POSITIONX#B,Y#B:PATTERN-B,MA#
1030 RETURN
1040 IFCHARACTER$(X,Y)=" " THENRETURN
1050 POSITIONX#B,Y#B:PATTERN-B,C#;Y=Y-1:POSITIONX#B,Y#B:PATTERN-B,MA#
1060 RETURN
1070 A#CHARACTER$(X,Y+1):IF(A#="/" )+(A#="\")GOTO1090
1080 RETURN
1090 POSITIONX#B,Y#B:PATTERN-B,C#;Y=Y-1:POSITIONX#B,Y#B:PATTERN-B,MA#
1100 RETURN
1110 A#CHARACTER$(X-1,Y)
1120 POSITIONX#B,Y#B:PATTERN-B,C#
1130 IF(A#="/" )+(A#="\")THENY=Y-1
1140 X=X-1:POSITIONX#B,Y#B:PATTERN-B,MA#:RETURN
1150 A#CHARACTER$(X+1,Y)
1160 POSITIONX#B,Y#B:PATTERN-B,C#
1170 IF(A#="/" )+(A#="\")THENY=Y-1
1180 X=X+1:POSITIONX#B,Y#B:PATTERN-B,MA#:RETURN
1190 REM
1200 IFCHARACTER$(X,Y+1)<>" " THENRETURN
1210 POSITIONX#B,Y#B:PATTERN-B,C#;Y=Y+1:POSITIONX#B,Y#B:PATTERN-B,MA#
1220 RETURN
1230 MUSIC"A1":S=S+100:RETURN
1231 FORI=0TO5:OUT0227,5:OUT0227,4:NEXT:S=S+100:RETURN
3000 MUSIC"B5R0-B4R0-B2R0-B4D4#C2#C4-B4-B2-B4-#A2-B7"
3300 MA=MA-1:IFMA=<0THEN10000
3320 GOTO60
5000 TEMPO7:FORD=0TO1: MUSIC"B6A3B6A3B+DBA#FE#FA":NEXT
5300 S=S+1000:N=N+1:IFN=5 THENL=L+1
5400 GOTO60
10000 CURSOR15,15:PRINT"GAME OVER!!!"
10010 IFH1<5THENH1=S
10020 CURSOR15,18:PRINT"PLAY AGAIN"
10030 CURSOR15,19:PRINT"(Y/N)":CR=CR-1:MA=3
10040 GETA#:IFA#="Y"THEN10070
10060 GOTO10040
10070 IFCR=<0THENGOTO49
10080 GOTO21000
20000 PRINTCHR$(6);:GRAPH1,C:GRAPH2,C
20010 PRINT" "
20020 PRINT" | 000 0000 0 0 0 0 0000 0 0 "
20030 PRINT" | 0 0 0 0 00 0 0 0 0 0 0 0 0 "
20040 PRINT" | 0 0 0 0 0 00 00 000 0 0 0 "
20050 PRINT" | 0 0 0 0 0 0 0 0 0 0 0 0 "
20060 PRINT" | 000 0000 0 0 0 0 0000 0 "
20070 PRINT" | "
20080 PRINT" | 0 0 0000 0 0 0000 "
20090 PRINT" | 0 0 0 0 00 0 0 "
20100 PRINT" | 00 0 0 0 00 0 00 "
20110 PRINT" | 0 0 0 0 0 0 0 0 "
20120 PRINT" | 0 0 0000 0 0 0000 0 0 0 0 "
20130 PRINT" | "
20140 PRINT" | コノ Program ハ コンテント"ドンキー コング
20150 PRINT" | "
20160 PRINT" | ラ ニヒテ ツワガ モノデ ソノナラ
20170 PRINT" | "
20180 PRINT" | 'DONKEY KONG 0 0 0 0 0 トイマス
20190 PRINT" | "
20200 PRINT" | INSERT COIN OYEN
20210 PRINT" | "
20220 PRINT" | ONE PLAY OYEN TWO PLAY 99999999YEN!"
20230 PRINT" | CREDIT 0 "
20235 PRINT" | "
20240 TEMPO7: MUSIC"F5GA3R+CR+C6+D3+CRARF6G3ARARG65F" :PRINTCHR$(5)
20250 GETA#:IFA#="0"THENRETURN

```

DONKEY KONG

BASICプログラム・リスト

```

20252 CURSOR32,22:PRINTCR;
20255 IFA$="." THENCR=CR+1:CURSOR32,22:PRINTCR;
20260 GOTO20250
21000 GRAPH1,C:PRINTCHR$(6):CURSOR15,10:PRINT" PUS (S)KEY":CURSOR26,22:PRINT" (
REDIT":CR
21010 GETA$:IFA$="S" GOTO50
21020 GOTO21010
30000 OUT3232,128:USR($C170):OUT3232,0
30010 GRAPH 11,01,C: POSITION19*8,B:PATTERN-8,G1$
30020 POSITION2*8,4*8:PATTERN-16,K0$
30030 POSITION8,21*8:PATTERN-16,H0$
30040 POSITION0,23*8:PATTERN-8,EA$
30045 CURSOR29,13:PRINT" "
30050 RETURN
30100 OUT3232,128:USR($C188):OUT3232,0
30110 GRAPH 11,01,C: POSITION19*8,B:PATTERN-8,G1$
30120 POSITION15*8,0:PATTERN-16,K0$
30125 CURSOR16,16:PRINT" "
30126 CURSOR12,6:PRINT" "
30127 CURSOR19,6:PRINT" "
30150 RETURN
30200 OUT3232,128:USR($C1A0):OUT3232,0
30210 GRAPH 11,01,C: POSITION7*8,B:PATTERN-8,G1$
30220 POSITION2*8,6*8:PATTERN-16,I0$
30221 CURSOR5,6:PRINT" "
30222 CURSOR29,6:PRINT" "
30223 CURSOR9,17:PRINT" "
30224 CURSOR29,19:PRINT" "
30225 CURSOR2,10:PRINT" "
30250 RETURN
30300 OUT3232,128:USR($C1B8):OUT3232,0
30310 GRAPH 11,01,C: POSITION20*8,B,24:PATTERN-8,G1$
30320 POSITION14*8,2*8:PATTERN-16,K0$
30325 CURSOR10,21:PRINT" "
30326 CURSOR20,21:PRINT" "
30327 CURSOR17,9:PRINT" "
30328 CURSOR14,17:PRINT" "
30350 RETURN
40000 G=LV: IFG<5 THENG=5:W=60-G*7
40001 E=INT(RND(1)*W): IFE=1 THENCURSOR25,5:PRINT" "
40010 E=INT(RND(1)*W/1.2): IFE=1 THENCURSOR23,2:PRINTCHR$(93)
40020 E=INT(RND(1)*W/1.2): IFE=1 THENCURSOR20,1:PRINTCHR$(93)
40030 E=INT(RND(1)*40): CURSORHX(E,1),HY(E,1):PRINT" H"
40050 OUT3232,128:USR($C000):OUT3232,0
40090 RETURN
40100 G=LV: IFG<5 THENG=5:W=70-G*10
40101 E=INT(RND(1)*40): CURSORHX(E,2),HY(E,2):PRINT" H"
40110 E=INT(RND(1)*W): IFE=1 THENCURSOR1,7:PRINT" "
40120 E=INT(RND(1)*W/1.2): IFE=1 THENCURSOR28,7:PRINT" "
40121 E=INT(RND(1)*W/1.2): IFE=1 THENCURSOR15,1:PRINT" "
40122 E=INT(RND(1)*W/1.7): IFE=1 THENCURSOR22,1:PRINT" "
40130 E=INT(RND(1)*35): IFE=1 THENCURSOR1,17:PRINT" "
40132 F=F+1
40135 IFF<>0 THEN40140
40136 OUT3232,128:USR($C100):OUT3232,0
40137 IFF=1 THENPOSITIONX*8,Y*8:PATTERN-8,C$:X=X+1:POSITIONX*8,Y*8:PATTERN-8,P
*
40138 IFF(Y=7)*(X<15) THENPOSITIONX*8,Y*8:PATTERN-8,C$:X=X+1:POSITIONX*8,Y*8:PAT
ERN-B,MA$
40139 IFF(Y=7)*(X>15) THENPOSITIONX*8,Y*8:PATTERN-8,C$:X=X-1:POSITIONX*8,Y*8:PAT
ERN-B,MA$
40140 IFF>1 THENF=-1
40190 RETURN
40200 G=LV: IFG<5 THENG=5:W=60-G*10
40201 E=INT(RND(1)*W/3): IFE=1 THENCURSOR1,4:PRINT" "
40210 OUT3232,128:USR($C300):USR($C100): OUT3232,0
40240 E=INT(RND(1)*40): CURSORHX(E,3),HY(E,3):PRINT" H"
40250 E=INT(RND(1)*W): IFE=1 THENCURSOR29,7:PRINT" "
40260 E=INT(RND(1)*W/1.2): IFE=1 THENCURSOR2,11:PRINT" "
40270 E=INT(RND(1)*W/1.7): IFE=1 THENCURSOR18,11:PRINT" "
40271 IFF(X=4)+(X=5) THENPOSITIONX*8,Y*8:PATTERN-8,C$:Y=Y-1:POSITIONX*8,Y*8:PATTE
RN-B,MA$
40272 CURSOR13,EL+9:PRINT" ";CURSOR4,23-EL:PRINT" ";EL=EL+1: IFFEL=>14 THEN
EL=0
40274 CURSOR13,EL+9:PRINT" / ";CURSOR4,23-EL:PRINT" \ ";
40290 RETURN
40300 G=LV: IFG<5 THENG=5:W=60-G*6
40301 E=INT(RND(1)*W/2): IFE=1 THENOUT3232,128:USR($C065):OUT3232,0
40330 E=INT(RND(1)*20): CURSORHX(E,4),HY(E,4):PRINT" H"

```

```

40335 E=INT(RND(1)*W/3):IF E<>1 THEN 40370
40340 E=INT(RND(1)*29):CURSOR,1:PRINT"▲"
40345 E=INT(RND(1)*29):CURSOR,1:PRINT"▲"
40370 IF CHARACTER$(X,Y+1)<>" " THEN 40380
40371 CURSOR,X,Y+1:PRINT" ":POKE52224+X+40*(Y+1),32:KU=KU-1:MUSIC"A1":S=S+100
40380 IF KU=0 THEN 50000
40390 OUT232,128:USR($C100):OUT232,0:RETURN
50000 FOR I=0 TO 23:CURSOR,1:PRINT" " ; NEXT CURSOR,5:PRINT"
"
50010 FOR J=20 TO 23:CURSOR,1:PRINT"/////////////////" ; NEXT
50020 GRAPH1,01,C
50030 POSITION4*8,6*8:PATTERN-8,MA$:POSITION6*8,6*8:PATTERN-8,G1$
50035 C1$=STRING$(CHR$(0),32)
50040 FOR I=5 TO 19:POSITION4*8,I*8:PATTERN16,K0$
50045 FOR U=0 TO 10:OUT227,5:OUT227,4:NEXT
50050 POSITION14*8,I*8:PATTERN16,C1$:NEXT
50060 POSITION14*8,20*8:PATTERN16,K0$
50065 TEMP06:MUSIC"G6E5F2G5CR1D2EFGF4ED-G2-A-G-F-E-DR2EFGAG3R0G4R0+C6GF4ED5C3R0
C6":TEMP07
50070 LV=LV+1:N=1:GOTO60
55000 GRAPH12,012,C
55001 FOR X=0 TO 30:FOR Y=0 TO 24:A$=CHARACTER$(X,Y):IF (A$=" /")+ (A$=" \") THEN POSITIONX
*8,Y*8:PATTERN-8,FY$
55002 NEXT Y,X:GRAPH1:RETURN
59999 PRINTCHR$(5):END
60000 PRINTCHR$(5):" " ;
60001 OUT232,128:FOR I=53248 TO 54248:A=PEEK(I):POKEI-5120,A:NEXT:OUT232,0
60002 FOR I=48128 TO 49128:IF PEEK(I)=0 THEN POKEI,32:NEXT
61000 REMERK:*****
61010 REMERK:*
61020 REMERK:* D O N K E Y K O N G *
61030 REMERK:*
61040 REMERK:*
61050 REMERK:* Made by K.H
61060 REMERK:* 1982 Jan 8 r
61070 REMERK:* ヨウゾウ K.M
61080 REMERK:* |
61090 REMERK:* | S.Uny
61100 REMERK:* |
61110 REMERK:* | VALZA
61120 REMERK:* |
61130 REMERK:* | MZ-80B
61140 REMERK:* |
61150 REMERK:* | SHARP
61160 REMERK:* | Z-80A
61170 REMERK:* |
61180 REMERK:* | SB-6520
61190 REMERK:* |
61200 REMERK:* | S.R.D
61210 REMERK:* |
61220 REMERK:* | DIS
61230 REMERK:* | ASSENBRER
61240 REMERK:* |
61250 REMERK:* | MONITOR
61260 REMERK:* | SB-1520
61270 REMERK:* |
61280 REMERK:* | Fuji
61290 REMERK:* | Cassette
61300 REMERK:* |
61310 REMERK:* | ヨウゾウ
61320 REMERK:* | テンゾウ
61330 REMERK:* |
61340 REMERK:* |
61350 REMERK:* | ヨウゾウ ヨウゾウ
61360 REMERK:* | CRACKER
61370 REMERK:* | By Y.M.O
61380 REMERK:* |
61390 REMERK:* | ヨウゾウ ヨウゾウ
61400 REMERK:* | ヨウゾウ
61410 REMERK:* |
61420 REMERK:* | Oh!
61430 REMERK:* | SUSANNA
61440 REMERK:* |
61450 REMERK:* @ VALZA | ヨウゾウ
61460 REMERK:* |
61470 REMERK:*****
65505 MUSIC"G6E5F2G5CR1D2EFGF4ED-G2-A-G-F-E-DR2EFGAG3R0G4R0+C6GF4ED5C3R0C6"

```

■ DONKY KONGモドキ マシン語プログラム・リスト

Bdd	+0	+1	+2	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sun
B010	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	IAC
B020	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	9B	I52
B030	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	IAC
B040	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	I00
B050	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	I14
B060	2F	2F	2F	2F	2F	2F	2F	2F	2F	20	20	20	20	9A	20	I01
B070	53	43	4F	52	45	20	20	9A	9A	20	20	20	20	20	20	I50
B080	20	20	20	20	20	20	48	20	20	20	20	20	20	20	20	I50
B090	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	I14
B100	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	IAC
B110	20	20	20	20	20	20	48	20	20	20	20	20	20	9A	20	IAC
B120	20	20	20	20	20	20	9A	9A	20	20	20	20	20	20	20	I14
B130	20	20	20	20	20	20	48	20	20	20	20	20	20	20	20	IAC
B140	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	I50
B150	9A	20	20	20	20	20	20	20	20	20	20	20	20	20	20	IAC
Sun	1D	13	AF	B2	A5	B0	EE	F6	73	F6	F6	F6	F6	FB	I3	I62

Bdd	00	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
B10	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
B110	48	49	20	20	20	20	20	9A	9A	20	20	20	20	20	20	20	20	1AC
B120	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	1AC
B170	20	48	20	20	20	20	9A	20	53	43	4F	52	45	20	20	20	9A	1F
B140	9A	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	1B2
B150	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	1B2
B160	20	20	20	20	20	20	20	9A	9A	20	20	20	48	20	20	20	20	11C
B170	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	100
B180	20	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	9A	11C
B190	9A	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	196
B200	9A	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	196
B180	40	41	4E	20	20	20	20	9A	9A	20	20	20	20	20	20	20	20	170
B1C0	20	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	140
B1D0	5C	5C	5C	5C	5C	5C	5C	9A	20	20	20	20	20	20	20	20	20	1E4
B1E0	9A	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	17A
B1F0	20	48	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	1A2
Sum	0E	49	FE	EB	96	04	37	09	9B	9B	A7	5A	33	B9	1D	10	10	

Add	row	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+Sum	
B20	20	20	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	1F4	
B10	20	20	20	20	20	20	20	20	20	4B	20	20	20	20	20	20	12B	
B20	20	20	20	20	20	20	20	9A	20	4C	45	56	45	4C	20	20	9A	1C8
B20	9A	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	17A	
B40	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	140	
B20	20	20	20	20	20	20	20	20	9A	20	2F	2F	2F	2F	2F	2F	1A8	
B20	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	1B4	
B20	20	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	9A	1F4
B20	9A	20	20	20	20	20	20	20	4B	20	20	20	20	20	20	20	20	1A2
B20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	9A	20	17A	
B20	20	20	20	20	20	20	20	20	20	9A	20	20	20	20	20	20	17A	
B20	4B	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	12B	
B20	20	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	9A	1F4
B20	9A	5C	5C	5C	5C	5C	5C	5C	5C	5C	20	20	20	20	20	20	17A	
B20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	9A	20	17A
B20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	1CC
Sum	00	93	5C	4B	5A	5A	C8	42	9A	A7	63	52	4A	1E	7D	7D	121	

[illegible]

Add	00	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	180
B400	97	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	181
B410	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	182
B420	98	98	98	98	98	98	98	95	9A	20	20	20	20	20	20	20	183
B430	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	184
B440	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	9A	185
B450	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	186
B460	2F	2F	2F	2F	2F	2F	2F	2F	20	20	20	20	20	20	9A	20	187
B470	53	4F	4F	52	45	20	20	9A	9A	20	20	20	20	20	20	20	188
B480	20	48	20	20	20	20	20	20	20	20	20	20	20	20	48	20	189
B490	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	20	9A	190
B4A0	9A	20	20	20	20	20	20	20	48	70	20	20	20	20	20	20	191
B4B0	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	192
B4C0	20	20	20	20	20	20	9A	9A	20	20	20	20	20	20	20	20	193
B4D0	20	48	20	20	20	20	20	20	20	20	20	20	20	20	48	20	194
B4E0	70	20	20	20	20	9A	20	20	20	20	20	20	20	20	20	9A	195
B4F0	9A	20	20	20	20	20	20	20	48	20	20	20	20	20	20	20	196
Sum	1E	67	82	65	B9	16	4F	72	55	65	65	65	65	45	27	1A	197

Ad00	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	+G	+H	+I	+J	+K	+L	+M	+N	+O	+P	+Q	+R	+S	+T	+U	+V	+W	+X	+Y	+Z																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
B500	4B	49	20	20	20	20	20	4B	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	2

add	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	12sum
B600	20	20	20	20	20	20	20	9A	9A	5C	5C	5C	5C	5C	5C	5C	I50
B610	2F	2F	2F	2F	2F	2F	2F	20	20	5C	5C	5C	5C	5C	5C	5C	I6D
B620	20	20	2F	2F	20	20	9A	20	4C	45	56	45	5C	20	20	20	9A I5F
B650	9A	20	20	20	20	20	20	20	48	20	20	20	20	20	20	20	I62
B660	20	20	20	20	20	20	4B	20	20	20	20	20	20	20	20	20	I63
B670	20	20	20	20	20	20	9A	9A	20	20	20	20	20	20	20	20	I64
B690	20	4B	20	20	20	20	20	20	20	20	20	20	20	20	4B	20	I50
B670	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	20	9A I54
B690	9A	20	20	20	20	20	20	20	4B	20	20	20	20	20	20	20	I62
B690	20	20	20	20	20	20	4B	20	20	20	20	20	20	20	20	20	9A I62
B690	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	I70
B6B0	20	4B	20	20	20	20	20	20	20	20	20	20	20	20	20	4B	I50
B6C0	20	20	20	20	20	20	9A	20	20	20	20	20	20	20	20	20	9A I54
B6C0	9A	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	I5F
B6E0	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	5C	I60
B6E0	20	20	20	20	20	20	20	9A	9A	20	20	20	20	20	20	20	I6F
Sum	E4	EC	67	9A	76	87	45	60	BC	65	26	15	1C	B4	76	4E	I1F

B700	20	+1	+2	23	+4	+5	+6	+7	B	+7	+A	+B	+C	+D	E	F	15un
B710	48	20	20	20	20	9A	20	20	20	B2	20	20	20	20	20	9A	17E
B720	9A	20	20	20	20	20	20	48	20	20	20	20	20	20	20	20	1A2
B730	20	20	20	20	20	20	20	48	20	20	20	20	20	20	20	9A	18E
B740	20	20	20	20	20	F6	20	20	20	20	20	20	20	20	20	20	162
B750	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	100
B760	48	20	20	20	20	20	9A	20	20	B1	20	20	20	20	20	9A	17D
B770	9A	20	20	20	20	20	48	20	20	20	20	20	20	20	20	20	1AD
B780	20	20	20	20	20	20	20	20	48	20	20	20	20	20	20	9A	182
B790	20	20	F2	20	20	20	20	9A	9A	2F	2F	2F	2F	2F	2F	2F	1FO
B7A0	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	180
B7B0	2F	2F	2F	2F	2F	2F	2A	20	F5	D4	4A	55	4D	50	2A	9A	1FC
B7C0	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	1BD
B7D0	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	98	1B1
B7E0	98	98	98	98	98	98	98	96	00	00	00	00	00	00	00	00	001
B7F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	000
Sum	B4	D3	16	D2	45	4F	CE	9F	DE	F1	C1	09	01	04	C9	6A	132

[illegible][illegible]

DONKEY KONG

マシン語プログラム・リスト

```

B9C0 2F 2F 2F 2F 20 20 20 20 20 20 20 20 20 2F 2F 2F 17B
B9D0 20 20 20 20 20 4B 9A 20 20 20 20 20 20 20 20 9A 11C
B9E0 9A 2F 20 20 20 20 20 4B 20 20 4B 20 20 20 20 20 1EB
B9F0 2F 2F 2F 2F 20 20 20 20 20 20 20 20 20 20 4B 9A 20 1CF

```

Sum 1D EF 0B AF 7B 70 FA AA D2 D0 6E CA BC 7F D2 33 CB 194

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
B800 20 20 20 20 20 20 20 20 20 9A 9A 4B 20 20 20 20 4B 144
B810 20 20 4B 20 20 20 20 20 20 20 4B 20 20 20 20 20 20 150
B820 20 20 20 20 20 4B 9A 20 4C 45 56 45 4C 20 20 9A 1F4
B830 9A 40 20 20 20 20 20 20 4B 20 20 4B 20 20 20 20 20 1F2
B840 20 4B 20 20 20 20 20 20 20 20 20 20 20 4B 9A 1CA
B850 20 20 20 20 20 20 9A 9A 4B 20 20 20 20 20 20 20 20 144
B860 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 150
B870 20 20 20 2F 2F 2F 9A 20 20 20 20 20 20 20 20 9A 121
B880 9A 4B 20 20 20 20 20 4B 20 20 4B 20 20 20 20 20 1F2
B890 5C 5C 20 20 20 20 20 20 20 20 4B 20 20 20 9A 20 11A
B8A0 4B 45 59 20 20 20 20 9A 9A 5C 5C 20 20 20 20 4B 110
B8B0 20 20 20 20 20 20 20 20 20 20 20 20 5C 5C 20 20 10C
B8C0 20 20 20 20 20 20 20 20 20 20 20 20 5C 5C 20 20 10C
B8D0 9A 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 1F2
B8E0 20 20 20 20 20 20 20 20 5C 5C 5C 4B 20 20 20 9A 20 156
B8F0 20 20 20 20 20 20 20 20 9A 9A 20 4B 20 20 20 20 5C 130

```

Sum D5 D9 B1 37 0F 37 6E 9C 50 7D 12 B1 6B 64 6E 22 192

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
B800 5C 5C 5C 5C 20 20 20 20 20 20 20 20 20 20 20 20 1F0
B810 20 20 20 5C 5C 5C 5C 9A 20 20 20 20 20 20 20 20 9A 10A
B820 9A 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
B830 20 20 20 20 20 20 20 20 9A 9A 4B 20 20 20 20 20 1A2
B840 F4 B4 F3 F5 20 20 20 9A 9A 4B 20 20 20 20 20 20 11C
B850 20 20 20 20 20 20 20 20 9A 9A 4B 20 20 20 20 20 10F
B860 2F 20 20 20 2F 2F 9A 20 20 20 20 20 20 20 20 9A 1B2
B870 9A 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
B880 20 20 20 2F 2F 2F 20 20 4B 20 20 20 20 4B 20 9A 1F7
B890 20 20 20 20 20 20 20 20 9A 9A 2F 2F 20 20 20 20 20 1E4
B8A0 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 12B
B8B0 2F 20 20 5C 5C 9A 20 20 20 20 20 20 20 20 20 20 12F
B8C0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1AD
B8D0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1B1
B8E0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1F3
B8F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 109

```

Sum 70 F1 B4 77 BD AB BF 4F D6 F1 DA EB 9B 0E AB 37 155

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
BC00 97 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1AC
BC10 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1B2
BC20 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 100
BC30 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 160
BC40 20 20 20 20 20 20 20 20 9A 20 20 20 20 20 20 9A 1F4
BC50 9A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 17A
BC60 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 9A 17A
BC70 52 43 4F 42 45 20 20 9A 9A 20 20 20 20 20 20 20 190
BC80 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 100
BC90 20 20 20 20 20 20 20 20 9A 20 20 20 20 20 20 20 9A 1F4
BCA0 9A 20 20 20 20 20 20 2F 2F 2F 2F 2F 2F 2F 2F 2F 101
BCB0 4B 2F 2F 2F 2F 2F 2F 2F 2F 20 20 20 20 20 20 9A 11A
BCD0 20 20 20 20 20 20 20 20 9A 9A 20 20 20 20 20 20 1F4
BCE0 20 20 20 20 20 20 20 20 4B 20 20 20 20 20 20 20 12B
BCF0 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 9A 1F4
BCD0 9A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 17A

```

Sum 36 A7 AF B2 A5 B0 B0 EB AA 05 05 05 05 05 69 1F8

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
BD00 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BD10 4B 49 20 20 20 20 20 20 9A 9A 20 20 20 20 5C 5C 142
BD20 8B 8B 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 8B 17C
BD30 5C 5C 5C 5C 20 20 20 9A 53 4F 4F 52 45 20 20 9A 1B4
BD40 9A 20 20 20 20 4B 20 20 20 20 4B 20 20 20 20 20 1CA
BD50 20 20 20 20 4B 20 20 20 20 20 4B 20 20 20 20 20 1A2
BD60 20 20 20 20 20 20 20 20 9A 9A 20 20 20 20 20 20 10C
BD70 20 20 20 20 4B 20 20 20 20 20 20 4B 20 20 20 20 150
BD80 20 20 4B 20 20 20 20 20 9A 20 20 20 20 20 20 20 9A 11C
BD90 9A 20 20 20 20 4B 20 20 20 20 20 20 20 4B 20 20 1CA
BDA0 20 20 20 20 4B 20 20 20 20 20 4B 20 20 20 20 20 9A 1CA
BDB0 4D 41 4E 20 20 20 20 9A 9A 20 20 20 20 2F 2F 2F 100
BDC0 8B 8B 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 8B 179
BDD0 2F 2F 2F 2F 20 20 20 20 9A 20 20 20 20 20 20 20 9A 130
BDE0 9A 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BDF0 4B 20 20 20 20 20 20 20 20 20 20 4B 20 20 20 20 9A 1CA

```

Sum 54 6B F9 D2 73 9B 4B 27 05 6E CA DC 0B BE 21 77 164

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
BE00 20 20 20 20 20 20 20 20 9A 9A 20 20 4B 20 20 20 20 11C
BE10 20 20 20 20 20 20 20 20 4B 4B 20 20 20 20 20 20 20 12B
BE20 20 20 20 4B 20 20 20 20 9A 4C 45 56 45 4C 20 20 9A 1F4
BE30 9A 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BE40 4B 20 20 20 20 20 20 20 20 20 20 20 4B 20 20 20 9A 1CA
BE50 20 20 20 20 20 20 20 20 9A 9A 20 5C 5C 5C 5C 5C 15C
BE60 8B 8B 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 8B 17C
BE70 5C 5C 5C 5C 5C 5C 20 9A 20 20 20 20 20 20 20 20 20 1A2
BE80 9A 20 20 20 20 20 20 20 20 20 20 20 4B 20 20 20 20 1CA
BE90 20 20 20 4B 20 20 20 20 20 20 4B 20 20 20 20 20 9A 1CA

```

```

BEA0 4B 45 59 20 20 20 20 9A 9A 20 20 20 20 20 20 4B 20 1A5
BEB0 20 20 20 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 11C
BEC0 20 4B 20 20 20 20 20 20 9A 20 20 20 20 20 20 20 20 11C
BED0 9A 20 20 20 20 20 20 4B 20 20 20 20 20 20 4B 20 20 1CA
BEE0 20 20 20 4B 20 20 20 20 20 20 4B 20 20 20 20 20 9A 1CA
BEF0 20 20 20 20 20 20 20 20 20 9A 9A 2F 2F 2F 2F 2F 2F 125

```

Sum 6B F4 87 1B A0 3C 9C 92 7B 0C BD 24 03 B7 DE 92 10A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
BF00 8B 8B 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 179
BF10 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 2F 179
BF20 9A 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BF30 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BF40 F4 B4 F3 F5 20 20 20 9A 9A 20 20 20 20 20 20 20 20 11C
BF50 20 20 20 20 20 20 20 20 20 4B 20 20 20 20 20 20 20 150
BF60 20 20 20 4B 20 20 20 20 9A 20 20 20 20 20 20 20 20 9A 1F2
BF70 9A 20 20 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 1A2
BF80 4B 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 9A 1CA
BF90 20 20 20 20 20 20 20 20 9A 9A 5C 5C 5C 5C 5C 5C 5C 16A
BFA0 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 16A
BFB0 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 16A
BFC0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1AD
BFD0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1AD
BFE0 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 9B 1AD
BFF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum 5B A7 BE C2 BD E7 9C F9 0B 5A 2A EA 6A 6D 99 FC 16D

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
C000 21 07 D3 06 00 00 00 00 7E 9C 3B 0C 3B 0C 55 0C 7E 107
C010 CD 5E 09 77 22 10 C2 11 2B 00 19 7E 7F 2F 20 09 179
C020 2A 10 C2 2B 10 C3 3B 0C 5C 5C 5C 5C 5C 5C 5C 5C 18A
C030 23 3A 10 C3 3B 0C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 18A
C040 05 C0 21 E7 D3 7E FE 10 0A 62 36 93 2B D6 D6 102
C050 D8 C3 45 C0 C9 5E 11 00 04 67 5E 52 D1 C9 5E 11 189
C060 00 04 19 D1 C9 21 E7 D3 7E FE 2F 20 04 36 5C 8B 10F
C070 06 FE 5C 5C 02 36 2F 2F 2F D6 D6 D6 D6 D6 D6 D6 D6 189
C080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 189
C090 21 00 C2 D6 00 23 3A 00 23 3A 00 23 3A 00 23 3A 189
C0A0 00 0E 12 CD 3A 0C B5 57 20 00 00 23 00 C2 3A 0C 18C
C0B0 67 20 05 21 01 C2 3A 0C B5 6F 05 21 02 C2 3A 0C 18C
C0C0 01 C2 37 20 05 21 01 C2 3A 0C B5 6F 05 11 CD 0A C0 18C
C0D0 57 20 05 21 04 C2 3A 0C B5 6F 05 00 00 00 00 00 00 189
C0E0 06 BB BA C9 00 00 00 00 00 00 00 00 00 00 00 00 17A
C0F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum E4 04 7F 2B DB 12 CE B0 D0 2B BE B1 B4 C1 E7 1E9

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
C100 21 E7 D3 00 00 00 00 7E FE 83 C2 3B C1 CD 55 C1 7E 1F8
C110 CD 5E 09 77 22 10 C2 11 2B 00 19 7E 7F 2F 20 09 17D
C120 2A 10 C2 2B 10 C3 3B 0C 5C 5C 5C 5C 5C 5C 5C 5C 102 18A
C130 23 3A 10 C3 3B 0C 5C 5C 5C 5C 5C 5C 5C 5C 5C 5C 102 18A
C140 05 C0 21 E7 D3 7E FE 10 0A 62 36 93 2B D6 D6 102 18A
C150 08 C3 45 C0 C9 5E 11 00 04 67 5E 52 D1 C9 5E 11 189
C160 00 04 19 D1 C9 21 E7 D3 7E FE 2F 20 04 36 5C 8B 10F 189
C170 21 00 C2 D6 00 23 3A 00 23 3A 00 23 3A 00 23 3A 11 189
C180 00 01 E8 05 ED B0 C9 00 21 00 84 11 00 D0 01 E8 18D
C190 02 ED B0 21 00 84 11 00 CC 01 E8 05 ED B0 C9 00 1A4
C1A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 12D
C1B0 00 01 E8 05 ED B0 C9 00 21 00 84 11 00 D0 01 E8 18C
C1C0 02 ED B0 21 00 84 11 00 CC 01 E8 05 ED B0 C9 00 1AC
C1D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 160
C1E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 160
C1F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum F6 EF 7D 4B CF A4 FE 37 9C 95 35 CB 26 31 22 CC 18B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
C200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 101
C210 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 10B
C220 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C230 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C240 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C250 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C260 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C270 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C280 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C290 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C2F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100

```

Sum CB D0 01 00 00 00 00 00 00 00 00 00 00 00 00 00 10C

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F 1Sum
C300 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 100
C310 21 A1 D0 7E 70 47 00 00 21 A2 D0 7E 70 47 00 00 18F
C320 21 A1 D0 7E 70 47 00 00 21 A3 D0 7E 70 47 00 00 18F
C330 21 C1 D0 7E 70 47 00 00 21 F4 D0 7E 70 47 00 00 18F
C340 21 CD D0 7E 70 47 00 00 21 A6 D0 7E 70 47 00 00 18F
C350 21 A7 D0 7E 70 47 00 00 21 D0 D0 7E 70 47 00 00 18F
C360 21 A7 D0 7E 70 47 00 00 21 D0 D0 7E 70 47 00 00 18F
C370 21 F4 D0 7E 70 47 00 00 21 D2 D0 7E 70 47 00 00 117

```

DONKEY KONG

マシン語プログラム・リスト

C380	21	AB	D0	7E	70	47	00	00	21	AC	D0	7E	70	47	00	00	1A7
C390	21	D5	D0	7E	70	47	00	00	21	FD	D0	7E	70	47	00	00	11E
C3A0	21	26	D1	7E	70	47	00	00	21	FE	D0	7E	70	47	00	00	171
C3B0	21	87	D0	7E	70	47	00	00	21	B0	D0	7E	70	47	00	00	1D5
C3C0	21	81	D0	7E	70	47	00	00	21	B2	D0	7E	70	47	00	00	1AF
C3D0	21	D8	D0	7E	70	47	00	00	21	03	D1	7E	70	47	00	00	12B
C3E0	21	2C	D1	7E	70	47	00	00	21	04	D1	7E	70	47	00	00	17E
C3F0	21	DD	D0	7E	70	47	00	00	21	B6	D0	7E	70	47	00	00	1DF
Sum	F5	1F	33	62	90	29	00	00	EF	B8	33	62	90	29	00	00	157
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	15Sum
C400	21	87	D0	7E	70	47	00	00	21	E0	D0	7E	70	47	00	00	1E5
C410	21	08	D1	7E	70	47	00	00	21	31	D1	7E	70	47	00	00	1B7
C420	21	59	D1	7E	70	47	00	00	21	81	D1	7E	70	47	00	00	128
C430	21	D1	D1	7E	70	47	00	00	21	21	D2	7E	70	47	00	00	141
C440	21	71	D2	7E	70	47	00	00	21	C1	D2	7E	70	47	00	00	182
C450	21	11	D3	7E	70	47	00	00	21	61	D3	7E	70	47	00	00	18D
Sum	C6	6B	EB	F4	A0	AA	00	00	C6	D5	E9	F4	A0	AA	00	00	1E2



MZ-80B

DONKEY CLIMBER

《ドンキークライマー》



CRAZY CLIMBERは仕事を終えた。しかし、CRAZYがCLAZYになっていたため、死んだあとも多くの人からけなされていた。そこで作者は、「これは危い!」と思いCRAZY CLIMBERの親友DONKEY CLIMBERを呼んだ。CRAZY CLIMBERは腕が強いが、DONKEY CLIMBERは足が強い。そこで山登りに挑んだ。頂上にとどりつくまでに何が起ころであらうか!

■ぱっくまん

このCLIMBERは顔などをちょっとやさつとぶつけても死にませんが、足を（一番重要な部分）をぶつけたらころがり落ちます（図3）。

障害物は6種類あり、図4の4つが決まって出てきます。2面目からJUMPER、6面目からACOVYNA。

BALL, SPEAR, BARRIER, RISERを切り抜け、HIGHが220ないし221になると場面が変わり、頂上に着くシーンが現われます。これはDEMOなので、だまっていましよう。頂上に着くと万才をして、CRAZY CLIMBERにも出てきた影のはしご部隊がはしごを降ろしてくれます。

これで一面クリアです。一面ごとに違った音楽が流れます。また13面をクリアした後は、ずっと13面が続きます（イ吉だと思わないでください）。

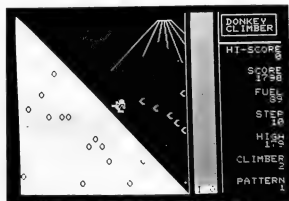
人間は一万点で1人増えます。エネルギーはある程度経つか、JUMPすると1ずつ減ります。人間が1人もいなくなるか、エネルギーが0になるとゲーム・オーバーです。

戦い終わって、また例のワンパターンで名前を書きます。いままでと違うところは、BGMがかつこよく、一連の音楽になっています。しかも、カーソルの移動する速度が少し早くなっています。また、**A**の方へ突っこむと**END**の方へ、**END**の方へ突っこむと**A**の方へ出ます。もちろん、5位以内にいないと名前は書けません。

ゲームの説明

このゲームのキー操作はちょっと難しいので、しっかり読んでください。操作は図1に示しますが、ここで一応説明しておきます。まず、行動範囲があり、この間を**7**と**3**のキーで動かします（図2-1参照）。**5**のSTEPというのは移動ではなく、“進む”ということです。早い話がこれによって、画面上スクロールして高さがあがり、得点があがります。

8のJUMPというのは、真上にある期間飛んでいるということです。このキーと**7**を同時に押すことにより、前にジャンプすることができます。このジャンプにより、行動範囲をはみ出したら、行動範囲に入るまですべり降ります（アーバン電子のK君がこの設定を聞いて突っ当）。



DONKEY CLIMBER

図 1

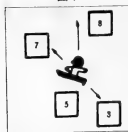


図 2-1



図 2-2



図 3

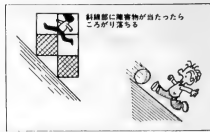


図 4

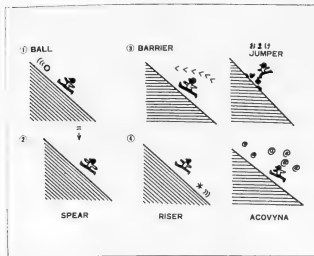


表 1 各ルーチンの処理

0~30	第1初期設定
40	第2初期設定& DEMO
45~90	スタート画面
95	スタートDEMO & MUSIC
100~590	メイン
600~660	人間が増えるか?
700~730	ジャンプ準備
800~880	ジャンプ
900~920	ステップ・サウンド
940~980	スタートのバックの山を置く
1000~1040	BALLの処理
1500~1540	RISERの処理
2000~2040	SPEARの処理
2500~2520	ACOVYNAの処理
3000~3040	BARRIERの処理
4000~4020	JUMPERの処理
5000~5290	面クリアDEMO
5300~5330	若者たち
5330~5370	お・スザンナ
5400~5440	楽しい家族
5550~5470	子もり歌(?)
5500~5530	一週間
5500~5570	ピクニック
5600~5620	森へ行きましょう
5650~5670	赤い川の谷間
5700~5720	旅愁
5750~5770	冬の星座
5800~5830	オールド・ブラック・ジョー
5850~5890	気球にのってどこまでも
5900~5930	面クリアの山
6000~6320	スタートの前のDEMO 1
6500~6530	スタートの前のDEMO 2
6900~6950	人が走って字を書くサブルーチン
7000~7090	ころがり落ちて死ぬ
7100~7200	ゲーム・オーバー
7200~7550	ネーミング
	字を一文ずつ出すサブルーチン
8000~8260	スタート画面の文字類
8500~8600	スタートDEMO
9000~9002	クリア・データ (パターン)
9005~9030	人間データ (パターン)
9100~9140	デモパターン・データ
9500~9520	マシン書き込み
9600~9620	MUSIC (NAMING) DATA変換
9700~9710	ネーミング・カーソル・データ
9750~9760	画面データ (デモ用)
9800	データ設定終了
9900	データ読み込みサブルーチン
10000~10280	2行ごとに違う人間のパターン・データ
15000~15020	(10260~10280だけ3行)
15000~15020	スクロール (マシン側)
15030	キー入力チェック (マシン側)
20000~	画面データ (DEMO用)

面クリアの音楽

初期データ
設定

プログラムの説明

あの一のあの一BASICです! でもそれがどうしたってんだ? I/Oプラザで読みやすいBASICのプログラムを期待してたんがいたじゃないか! ゲームなんていうものは、速度さえ落ちなければいいんだ!

マシン語が何だ! ほとんど読者に理解できないじゃないか! また、アセンブル・リストで載せるのならともかく、ダンブ・リストならマイコンをゲームだけの目的に変えているようなものじゃないか! その点BASICはみんなに理解できて、手軽に改造できるじゃないか! ...以上つまらないことで興味がスミマセン!

僕の場合、増殖が簡単にできるBASICを受してるんです? 見やすいように処理がはつきり別れているところにラインを入れています。どこで何をするかは、表1を参照してください。

一部マシン語を使っていますが、スクロールと同時にキーを押しているかを判断するサブルーチンです。全体のフローチャートは図5を見てください。

必勝法

■ BALL (球)

近くに引きつけておいて、前なし上にはジャンプする。

■ SPEAR (やり)

真上から落ち始めたら [3] と [7] で位置を変えて進む。

■ BARRIER (バリア)

進んでいる間はバリアが張られるので、地面に接近したら止まり、新しいバリアが出てくるまで待つて進む。

■ RISER (登り)

ぎりぎりまで引きつけて、真上に飛ぶ。当分進まないことがあるので注意。

■ JUMPER (ジャンプ台)


```

0 REM
0 REM DONKEYCLIMBER
0 REM
0 REM
0 REM 1982.1.23 MADE BY PUCK MAN
0 REM
0 REM ---Puck Man's ORIGINAL---
0 REM No. 12
0 REM
10 POKE$952,0:DEF KEY(5)=POKE$952,1667
12 LIMIT$=CEFF
14 TEMP07:DIMHU$(14),HS(5),HS$(5),UN$(7),G(4),CX(30),CY(30),TT$(20)
15 MU$="EEFGGFEDCCDEEDDEEFEGGFEDCCDEEDCCRDDECFEDCFEDCD-EEEEFGGFEDCCDEEDCCRR"
20 CONSOLE$0,24,N,C40:GRAPHIC,01
25 CURSOR20,23:PRINT"Made By Puck Man"
30 GOSUB9000
40 SC=0:PA=1:MC=3:MU=0:MF=1:GOSUB6000:GOSUB6500
45 DF=1:PRINTCHR$(6);
50 GRAPHIC:FORI=0TO15:LINE0,I,199-I,199:NEXT
51 CURSOR0,0
55 FORI=1TO16:PRINT:PRINTSTRING$(CHR$(30),I):NEXT:IFDFTHENFORI=1TO8:PRINT:PRINTSTRING$(CHR$(30),25):NEXT
56 IFDF=0THENFORI=1TO8:PRINT:PRINTSTRING$(CHR$(30),16+I):NEXT
57 LINE239,0,239,199:LINE232,0,232,199:LINE200,0,200,199:LINE207,0,207,199
58 PRINTCHR$(5);:FORI=1TO24-PA:PRINTTAB(26);STRING$(CHR$(31),3):NEXT
59 PRINTTAB(26);STRING$(CHR$(31),3):FORI=1TOPA:PRINT:PRINTTAB(26);"U":CHR$(30)
;"U":NEXT
60 GOSUB9040
80 IFDFTHENA=17:ST=0:XA=0:FU=100:SP=0:JF=0:S1=0:TS=0:FF=1
81 G(0)=INT((PA+6)/4)
82 G(1)=INT((PA+5)/4)
83 G(2)=INT((PA+4)/4)
84 G(3)=INT((PA+3)/4)
90 GOSUB8000
95 IFDFGOSUB8500
99 REM
100 F1=F1+1:W=0:B=A:GETA$:IFJF=1THENB00
105 IFA<14THENA=A+1:W=1
110 IFA$="7" THENA=A-1:A=A-(A<14):W=1
120 IFA$="3" THENA=A+1:A=A+(A>20):W=1
130 IF(A$="B")*(A>13) THENGOSUB700
140 IFA$<"5" GOT0200
145 IFSP=4THENCURSORSXB+1,XB:PRINT" ";
150 ST=ST+1:XA=XA+1
155 IFST>SP*50+20THENS1=1
156 IF(S1=1)*(SP=0)THENS=SP+1:S1=0:XA=0:XB=0
160 FF=FF+1:FF=FF+(FF=4)*4
165 IFSP=2THENCURSORSX,XB:PRINT" "
170 USR(%CF00):CURSOR0,0:PRINT" "
172 IFSP=2THENCURSORSX,XB:PRINT"4"
173 IFSP=4THENCURSORSXB+1,XB:PRINT"8";
175 CURSOR0,RND(1)*24+1:PRINT"U";
180 SC=SC+PA*10:GOSUB600
185 CURSOR38-LEN(STR$(SC)),9:PRINTSC
186 CURSOR38-LEN(STR$(ST)),18:PRINTST
190 ONFF+1GOSUB900,920,910
200 IFWTHENPOSITIONB*B,B*B-16:PATTERN-16,CL$
210 POSITIONA*B,A*B-16:PATTERN-16,HU$(FF)
215 CURSOR38-LEN(STR$(FU)),12:PRINTFU
220 C=ASC(CHARACTER$(A+1,A))+ASC(CHARACTER$(A,A-1))
230 IF(C<>64)*(C<>32)*(C<>0)THEN7000
250 IFF1:50THENFU=FU-1:F1=0
260 IFFU<1THENMC=1:GOTO7000
500 ONSPGOTO1000,2000,3000,1500,5000
590 GOTO100
599 REM
600 IF(SC>=10000)*MFTHEN620
610 RETURN
620 MF=0:MC=MC+1
630 MUSIC="CODEFGAB+C"
650 CURSOR38-LEN(STR$(MC)),21:PRINTMC
660 RETURN
699 REM
700 MUSIC="-D0":JF=1:J=A:FU=FU-1
710 USR(%CF60)
720 IFPEEK(%CFFF)=0THENJM=-1:JJ=0:RETURN
730 JM=-2:JJ=1:RETURN
799 REM
800 ONJJGOTO860
802 BJ=J:J=J+JM:IFJ/A=3THENJM=1
805 IFJ=ATHENJF=0
810 POSITIONA*B,BJ*B-16:PATTERN-16,CL$
820 POSITIONA*B,J*B-16:PATTERN-16,HU$(4)
830 MUSIC="-D0"
840 C=ASC(CHARACTER$(A+1,J))+ASC(CHARACTER$(A,J-1))

```

```

850 GOTO230
860 BJ=J:J=J+JM:A=A-1:IFJ<A-2THENJM=1
865 IFJ=A+1THENJF=0:J=J-1
870 POSITIONA#B#B,BJ#B-16:PATTERN-16,CL#
880 GOTOB20
899 REM
900 MUSIC"+B0":RETURN
910 MUSIC"+#B0":RETURN
920 RETURN
930 REM
940 LINE200,24,184,8,168,8,113,63
950 LINE170,8,150,48:LINE173,8,168,32
960 LINE176,8,180,40:LINE181,8,200,36
970 LINE178,8,196,56
980 RETURN
999 REM
1000 CURSORYA+1,XA:PRINT" ";
1010 XA=XA+1:IFXA>27THENXA=0
1020 CURSORXA+1,XA:PRINT"0";
1030 IF$1THENSP=SP+1:S1=0:XA=0:XB=0:XJ=0
1040 ONG(0)GOTO100,4000:GOTO2500
1050 REM
1500 XA=25:IFXB>27THENXB=23
1505 CF=2:CURSORXB+1,XB:PRINT" ";
1510 XB=XB-INT(RND(1)*1.5+.5):IFXB<1THENXB=23
1520 CURSORXB+1,XB:PRINT" ";
1530 IF$1THENSP=SP+1
1540 ONG(3)GOTO100,4000:GOTO2500
1550 REM
2000 IFXB>27THENXB=0
2005 XA=0:CURSORXJ,XB:PRINT" "
2010 XB=XB+1:IFXB>XJ+1THENXJ=INT(RND(1)*7)+14:XA=25:XB=0
2020 CURSORXJ,XB:PRINT"4"
2030 IF(S1=1)*(XA>23)THENSP=SP+1:S1=0:XA=0:XB=25:XJ=0
2040 ONG(1)GOTO100,4000
2050 REM
2500 IFRND(1)>.3THEN2520
2510 CURSORRND(1)*25,RND(1)*25:PRINT"@";
2520 ONG(SP-1)-2GOTO100,4000
2525 GOTO100 @ @
2530 REM
3000 IFXB>23THENXB=23
3005 CURSORXB,XJ:PRINT" ";
3010 XB=XB-INT(RND(1)*2):XB=XB+(XB>23):IFXB<=XJTHENXJ=INT(RND(1)*12):XB=24
3020 CURSORXB,XJ:PRINT"<";
3030 IF(S1=1)*(XB>22)THENSP=SP+1:S1=0:XA=0:XB=0
3040 ONG(2)GOTO100,4000:GOTO2500
3050 REM
4000 IFST-TS<10GOTO100
4010 IFRND(1)<.2THENCURSOR0,0:PRINT"0":CURSOR2,1:PRINT"0":CURSOR2,2:PRINT"0":T
S=ST
4020 GOTO100
4030 REM
5000 MUSIC"C2DEFGAB+CC1DEFGAB+CC0DEFGAB+CCDEFGAB+C
5010 PRINTCHR$(6):GRAPHIC
5020 LINE0,199,80,80,80,199
5030 LINE319,199,239,80,239,199
5040 FORI=B0T0192STEPB:LINEB0,I,239,I:NEXT:GOSUB5900
5050 FF=0
5060 FORI=184T0B0STEP-B
5070 POSITION152,I:PATTERN-16,HU$(10+FF)
5080 FORJ=1T0200:NEXT
5090 POSITION152,I:PATTERN-16,HU$(12)
5100 FF=1-FF:NEXT
5110 POSITION152,72:PATTERN-16,HU$(13)
5200 FORI=0T09:CURSOR18,I:PRINT" | ";FORJ=1T0200:NEXTJ,I
5210 FORI=10T01STEP-1:CURSOR18,I:PRINT" ";POSITION152,80:PATTERN-16,HU$(12)
5220 POSITION152,I#B-8:PATTERN-24,HU$(14):FORJ=1T0100:NEXT
5230 POSITION152,I#B-8:PATTERN-24,CC#NEXT
5240 CURSOR18,0:PRINT" ";
5260 SC=SC+FU*PA#10:PA=PA+1:PA=PA+(PA>13)
5265 GOSUB600
5270 ONPA-1GOSUB5300,5350,5400,5450,5500,5550,5600,5650,5700,5750,5800,5850
5280 FORI=1T01000:NEXT
5290 GOTO45
5299 REM
5300 MUSIC"C5E68G5G7E5DC9R7C5E68A5G7F5ED9R7
5310 MUSIC"E5DC9R7GE9R7E5DC8C5C7+C+C9R7
5320 MUSIC"A5B+CB+C5+C7AG9R7C5DE8E5DE7D5C9R7
5330 RETURN
5340 REM
5350 MUSIC"C3DE5GGAGEC6D3E5EDCD8C3DE5GGAGEC6D3E5EDDC9
5360 MUSIC"F7FA5A7A5GGECDC8C3DE5GGAGEC6D3E5EDDCB
5370 RETURN
5380 REM
5400 MUSIC"-B3D5D3D7C3E5E3E7#F5#FE#FGAB7
5410 MUSIC"-B3D5D3D7C3E5E3E7#F5#FE#FAGG7

```

DONKEY CLIMBER

プログラム・リスト

```

5420 MUSIC" B5GD7C5EE7#F3#F5#F3E#FGAB7
5430 MUSIC" B5GD7C5EE7#F3#F5#F3E#FGG67
5440 RETURN
5445 REM
5450 MUSIC" B8B5+D9AB85+C9B7BA5G#FGA9D
5460 MUSIC" B8B5+D9AB85+C9B7BA5B+CAG9R
5470 RETURN
5480 REM
5500 MUSIC" E5GB7B5ABBBABAG7R85AB7A5GAAG#FG#FE7R
5520 MUSIC" E5GBBBABBBABAG7R85ABBAGA7B+E9R7
5530 RETURN
5540 REM
5550 MUSIC" D5G7A5B7GE5G7E5D9-B5D7E5G7#F5G+D9B
5560 MUSIC" D5G7A5B7GE5G7E5D9+D5B7+D5+CA7B5G9R
5570 RETURN
5580 REM
5600 MUSIC" D7G5B7A5G7#F5G#FED7D5-B7C5D7B5G7D5C7C5-A7-B5C7A5#F7E5D7D5-B7C5DBAG#F
E
5610 MUSIC" D7G5B7A5G7#F5G#FED7D5-B7C5D7B5G7D5C7C5-A7-B5C7A5#F7E5D7D5B7A5G9
5620 RETURN
5630 REM
5650 MUSIC" -G5CE7E5EE7D5EDCB7R-G5CE7C5EG7F5ED9R7
5660 MUSIC" G5FE7E5DC7D5EGFBR7-A5-#G-G7-B5CD7E5DC9R7
5670 RETURN
5680 REM
5700 MUSIC" G7E5G+C9A7+C5+CG9G7C5D7D5CD9R
5710 MUSIC" G7E5G+C8B5A7+CG9G7D5EF8-B5C9R
5720 RETURN
5730 REM
5750 MUSIC" C8D5EGA+CG9E7RC8D5EAGED9R
5760 MUSIC" C8D5EGA+CG9E7RAB85+CEDEC9R
5770 RETURN
5780 REM
5800 MUSIC" C7E6F3G7R5G3GA5+CBA9
5805 MUSIC" C7E6F3G7R5G3GA5GFAE3D9
5810 MUSIC" C7E6F3G7R5G3GA5+CBA97R5
5815 MUSIC" +C5B6+C3+D5B+CAGAE7DC8
5820 MUSIC" B5EGRGEGRG3GA5+CBA97R5
5825 MUSIC" +C5B6+C3+D5B+CAGAE7DC8
5830 RETURN
5840 REM
5850 TEMPO6
5855 MUSIC" R3EEDEREDERR5REA7A5ED9R3FFEFREFERR5R+C8A3GE7R5
5860 MUSIC" E3F05G7E3FG8A5G6E3C5CAA7F3GABF3GABA5F6E3D5DBB7
5870 MUSIC" A3G+C5+C6+C3+D+C8B+C3BA6A5A5B3AB8
5880 MUSIC" G5AA6A3B+C+D8+D5+E7+DR5+D+CB+CBB3A+CBB3A+C7+D+C9R
5890 TEMPO7:RETURN
5899 REM
5900 LINE223,80,319,0,290,40:LINE294,60,319,0,310,30
5910 LINE0,40,24,40,74,90
5920 LINE4,40,4,80:LINE7,40,10,75:LINE14,40,30,85:LINE18,40,60,98
5930 RETURN
5940 REM
6000 PRINTCHR$(6):GRAPHIC:CONSOLE0,22:F=0:DN=1
6005 CURSOR20,23:PRINT"START ANY KEY."
6010 S$=STRING$( "",12):M$=S$+"DONKEY CLIMBER"+S$
6020 T=20:S=0:GOSUB6900:IFF=1THENRETURN
6030 FORI=1TO17:CURSOR0,22:PRINT:GETA$:IFA$<>" "THENRETURN
6035 FORJ=1TO50:NEXTJ,I
6040 CURSOR12,2:PRINT" "
6050 CURSOR12,3:PRINT" |DONKEY CLIMBER|"
6060 CURSOR12,4:PRINT" "
6070 CURSOR4,8:PRINT" " :PRINT" " :PRINT" "
6080 CURSOR10,14:PRINT" " :PRINTTAB(10):" | " :PRINTTAB(10):" "
6085 FORI=38TO5STEP-1:CURSORI,9:PRINT"7":GETA$:IFA$="" THENFORJ=1TO50:NEXT:CURSORI,9:PRINT" " :NEXT:GOTO6087
6086 RETURN
6087 CURSOR4,8:PRINT" " :PRINT" " :PRINT" " :PRINT" "
6090 FORI=38TO1STEP-1:CURSORI,15:PRINT"3":GETA$:IFA$="" THENFORJ=1TO50:NEXT:CURSORI,15:PRINT" " :NEXT:GOTO6092
6091 RETURN
6092 CURSOR10,14:PRINT" " :PRINTTAB(10):" |3| " :PRINTTAB(10):" "
6100 M$="" :T=12:S=7:GOSUB6900
6105 IFF=1THENRETURN
6110 CURSOR7,11:PRINT" " :PRINTTAB(7):" |5| -STEP " :PRINTTAB(7):" "
6120 FORI=1TO200:GETA$:IFA$="" THENNEXT:GOTO6130
6125 RETURN
6130 FORI=12TO10STEP-1:POSITION60,I#8-4:PATTERN-16,HU$(4):GETA$:IFA$="" THENFORJ=1TO200:NEXT:POSITION60,I#8-4:PATTERN-16,CL$:NEXT:GOTO6140
6135 RETURN
6140 POSITION60,68:PATTERN-16,HU$(4)
6150 CURSOR7,8:PRINT" " :PRINTTAB(7):" |8| -JUMP " :PRINTTAB(7):" "
6160 FORI=1TO200:GETA$:IFA$="" THENNEXT:GRAPHIC:GOTO6170
6165 RETURN
6170 CURSOR17,8:PRINT"ヒダリノ KEY 9 CLIMBER9"
6180 PRINTTAB(17):"フコ カシテ ヒダリ ノキ ヲマダシ ヲイ、"
6190 PRINT:PRINTTAB(17):"フコ 220m ノキ ヒダリ ヲマダシ ヲイ、"

```

```

6200 PRINT:PRINTTAB(17);"  ~~~ ショウワ イアツ ~~~
6210 PRINT:PRINTTAB(17);"  0  ----- BALL
6220 PRINT:PRINTTAB(17);"  4  ----- SPEAR
6230 PRINT:PRINTTAB(17);"  <  ----- BARRIER
6240 PRINT:PRINTTAB(17);"  *  ----- RISER
6250 FORI=1TO1000:GETA$:IFA$="" THENNEXT:GOTO6260
6255 RETURN
6260 PRINTCHR$(6);TAB(14);"HI-SCORES"
6270 FORI=0TO4:CURSOR0,2*I+2:PRINTTAB(16-LEN(STR$(HS(I)))):HS(I);"  ";HS(I):NE
XT
6280 FORI=1TO1000:GETA$:IFA$="" THENNEXT:GOTO6290
6285 RETURN
6290 PRINTCHR$(6);FORI=0TO20:M$=TT$(I):T=I+1:S=0:GOSUB6900:IFF=1THENRETURN
6300 NEXT
6310 FORI=1TO1000:GETA$:IFA$="" THENNEXT:PRINTCHR$(6):GOTO6010
6320 RETURN
6330 REM -----
6500 DM=0:CONSOLE0,24:GRAPHIC
6510 H$="  PUSH (8) KEY FOR GAME START":T=10:S=0:GOSUB6900:IFAA=1THEN6510
6530 PRINTCHR$(6);:RETURN
6899 REM -----
6900 HF=2:AA=0:FORD=38TO1STEP-1
6905 IFS=0THENRETURN
6910 HF=HF+1:IFHF=7THENHF=3
6920 POSITION0*8-4,T*8-4:PATTERN-16,UN$(HF):CURSORQ,T:PRINTMID$(M$,Q,1);
6925 GETA$:IF(A$<"")$DMTHENF=1:RETURN
6926 IFA$="S" THENRETURN
6930 NEXT:AA=1
6940 POSITION0*8+4,T*8-4:PATTERN-16,CL$
6950 RETURN
6960 REM -----
7000 IFJF=1THENPOSITIONA*8,J*8-16:PATTERN-16,CL$
7005 P=5:FORI=ATO23
7010 POSITIONI*8,I*8-16:PATTERN-16,HU$(P)
7020 P=P+1:IFP=9THENP=5
7030 MUSIC"C2"
7035 POSITIONI*8,I*8-16:PATTERN-16,CL$
7040 NEXT
7050 POSITION192,176:PATTERN-16,LEFT$(HU$(P),16)
7060 MUSIC"C":C$=STRING$(CHR$(0),16)
7070 POSITION192,176:PATTERN-16,C$
7080 MUSIC"-B3-G-E-A-F-D-C5RO
7090 MC=MC-1:IFMC>0THENA=17:DF=0:JF=0:XA=0:XB=23:PRINTCHR$(6):GRAPHIC:GOTO50
7095 REM -----
7100 MUSIC"RBR":PRINTCHR$(6);:GRAPHIC
7110 CURSOR15,10:M$="GAME OVER":GOSUB7900
7120 FORI=1TO1000:NEXT
7130 CURSOR20,23:M$="Made By Puck Man":GOSUB7900
7140 FORI=1TO3000:NEXT
7150 FORI=4TO0STEP-1:IFHS(I)=<SCTHENNEXT:00=0:GOTO7170
7160 00=I+1:I=-1:NEXT
7170 FORI=4TO0STEP-1:HS(I+1)=HS(I):HS*(I+1)=HS*(I):NEXT
7180 HS(00)=SC:HS(00)=":PRINTCHR$(6);
7190 FORI=0TO4:CURSOR0,2*I+2:PRINTTAB(16-LEN(STR$(HS(I)))):HS(I);"  ";HS(I):NE
XT
7200 IFHS(4)>SCTHENFORI=1TO3000:NEXT:GOTO40
7205 REM -----
7210 FORI=10TO19:C$=CHARACTER$(I,00*2+2):IF(C$="0")$(C$="9")THENCURSORI,00*2+
2:PRINTCHR$(ASC(C$)+192);
7220 NEXT:IN$="  ":A=1:MUSIC"R4"
7230 CURSOR5,12:PRINT"A B C D E F G H I J
7240 CURSOR5,14:PRINT"K L M N O P Q R S T
7250 CURSOR5,16:PRINT"U V W X Y Z  . RUBEND
7260 CURSOR10,20:PRINT"NAME:"CURSOR10,21:PRINTSTRING$(IN$,21)
7270 CURSOR15+LEN(IN$),21:C$=CSRH:PRINT"  ";
7280 B=0:GETA$:IFA$="" GOTO7340
7290 IFA$="4" THENB=A:A=A-1:IFA=0THENA=30
7300 IFA$="6" THENB=A:A=A+1:IFA=31THENA=1
7310 CURSORC(A),CY(A):PRINT"  ";
7320 CURSORC(B),CY(B):PRINT"  ";
7330 GOTO7410
7340 IFA=30THEN7460
7350 IF(A=29)$(LEN(IN$)>0)THENIN$=LEFT$(IN$,LEN(IN$)-1):GOTO7410
7360 IFA=29THEN7410
7370 IFLEN(IN$)=16THEN7410
7380 IFA=28THENIN$=IN$+"  ":GOTO7410
7390 IFA=27THENIN$=IN$+"  ":GOTO7410
7400 IN$=IN$+CHR$(A+64)
7410 CURSOR10,20:PRINT"NAME:"IN$;"  ";
7420 IFMU=65THENMU=0
7430 MU=MU+1
7440 MUSICMID$(MU,MU*2-1,2)
7445 IFC<31THENCURSORC,21:PRINT"~";
7450 GOTO7270
7460 HS(00)=IN$:PRINTCHR$(6);
7470 FORI=0TO4:CURSOR0,2*I+2:PRINTTAB(16-LEN(STR$(HS(I)))):HS(I);"  ";HS(I):NE
XT

```

DONKEY CLIMBER

プログラム・リスト

```

7480 FORI=10T019:C%=CHARACTER$(I,00*2+2):IF(C%="0")*(C%<="9")THENCURS0R1,00*2+
2:PRINTCHR$(ASC(C%)+192);
7490 NEXT
7500 FORI=1T020:FORJ=1T0120:NEXT
7510 IFMU=65THENMU=0
7520 MU=MU+1
7530 MUSICHID$(MU,MU*2-1,2)
7540 NEXT
7550 GOTO40
7560 REM
7900 FORI=1T0LEN(M$):PRINTHID$(M$,I,1):FORJ=1T0100:NEXTJ,I:RETURN
7999 REM
8000 PRINTCHR$(5);
8010 PRINTTAB(30);"
8020 PRINTTAB(30);" | DONKEY |
8030 PRINTTAB(30);" | CLIMBER |
8040 PRINTTAB(30);"
8050 PRINTTAB(30);"
8060 PRINTTAB(30);" HI-SCORE
8070 PRINTTAB(38-LEN(STR$(HS(0)))):HS(0)
8080 PRINT
8090 PRINTTAB(30);" SCORE
8100 PRINTTAB(38-LEN(STR$(SC))):SC
8110 PRINT
8120 PRINTTAB(30);" FUEL
8130 PRINTTAB(30);" 100
8140 PRINT
8150 PRINTTAB(30);" STEP
8160 PRINTTAB(37-LEN(STR$(PA))):PA*10
8170 PRINT
8180 PRINTTAB(30);" HIGH
8190 PRINTTAB(38-LEN(STR$(ST))):ST
8200 PRINTTAB(30);"
8210 PRINTTAB(30);" CLIMBER
8220 PRINTTAB(38-LEN(STR$(MC))):MC
8230 PRINT
8240 PRINTTAB(30);" PATTERN
8250 PRINTTAB(38-LEN(STR$(PA))):PA;
8260 RETURN
8270 REM
8500 FORI=0T07
8530 POSITION192-I*8,120:PATTERN-16,UN$(I)
8540 FORJ=1T0200:NEXTJ,I
8550 FORI=0T03
8560 POSITION136,120:PATTERN-16,MU$(I)
8570 USR$(CFOO):CURS0R0,0:PRINT " ";
8580 FORJ=1T0200:NEXTJ,I
8590 MUSIC"C2DEFGFCDEFGAGFDEFGAGFDEFGEDCEGEDEFGAAAAAGFEFGFEDCD7
8600 RETURN
8699 REM
9000 CL%=STRING$(CHR$(0),32)
9002 CC%=STRING$(CHR$(0),48)
9003 REM -----
9005 RESTORE10000
9010 FORR=1T013:GOSUB9900:HU$(R)=I$:NEXT
9020 HU$(0)=HU$(1):HU$(1)=HU$(3)
9030 I$="":FORI=0T047:READY:I$=I$+CHR$(Y):NEXT:HU$(14)=I$
9040 REM -----
9100 UN$(0)=LEFT$(HU$(9),16):C%=STRING$(CHR$(0),16)
9110 UN$(1)=HU$(0):UN$(2)=HU$(9)+C%
9120 UN$(3)=HU$(2)+C%:UN$(4)=UN$(2)
9130 UN$(5)=UN$(1)+C%:UN$(6)=UN$(2)
9140 UN$(7)=UN$(3)
9150 REM -----
9500 RESTORE15000
9510 FORI=0T089:READY:POKE52992+I,Y:NEXT
9520 FORI=0T022:READY:POKE53088+I,Y:NEXT
9530 REM -----
9600 M1$="":FORI=1T066:IFHID$(MU$,I,1)="-" THENI=I+1:M1$=M1$+" "+MID$(MU$,I,1):N
EXT:GOTO9620
9610 M1$=M1$+" "+MID$(MU$,I,1):NEXT
9620 MU$=M1$
9630 REM -----
9700 FORI=1T010:FR=I*3+1:CX(I)=FR:CX(10+I)=FR:CX(20+I)=FR:NEXT
9710 FORI=0T02:FORJ=1T010:CY(I*10+J)=I*2+13:NEXTJ,I
9750 RESTORE20000
9760 FORI=0T020:READY:TT$(I)=Y$:NEXT
9800 RETURN
9899 REM -----
9900 I$="":FORI=0T031:READY:I$=I$+CHR$(Y):NEXT:RETURN
9999 REM
10000 DATA0,0,0,7,7,5,134,255,255,15,7,3,1,1,1,0
10010 DATA0,188,110,126,126,255,65,157,21,39,199,192,192,252,252,4
10020 DATA0,0,0,7,7,5,134,225,253,63,7,3,3,1,1,0
10030 DATA0,188,110,126,126,207,9,205,23,167,192,128,192,252,252,4
10040 DATA0,0,0,0,0,0,1,2,3,7,31,61,115,7,30,126

```


プログラム・リスト

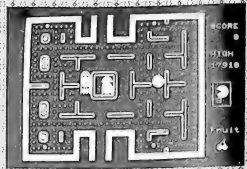
305

MZ-80B

PUCK MAN

パックマン

パックマンは、インベーダー、ギャラクシアンに次ぐ、ヒット作品と言えると思います。出現してから、かなりたちますが、まだゲーム・センターでは根強い人気があるようです。このゲームをMZ-80Bでも楽しもうと作ってみました。キャラクターで作れば楽なのですが、これではせつかくの320×200ドットが泣くので、グラフィックを使ってできるだけ本物に近くしてみました。



■ルリクデバ

(注) このゲームには、グラフィックV-RAM1が必要です。

1 ゲームについて

有名なゲームですから知っていると思います。ようすろにドットをすべて食べてしまえばいいのです。モンスターは4匹いて、まん中の部屋から次々と出て来てパックマンを追いかけます。同じ形なのと区別はつきにくいと思いますが、この4匹はそれぞれ違った追いかけ方をします。パックマンの移動はテン・キーの[2]、[4]、[6]、[8]です。キーを押しているとその方向に動き、離すとそこに止まっています。四隅に大きな丸が4つありますが、これはパワーエサで、パックマンがこれを食べるとモンスターはいもけて黒くなり、弱虫になって逃げまわります。このときはモンスターに食いつくことができ、連続して食べていくと得点は200、400、800、1600と増えていきます。

ときどき、中央の通路にフルーツが出てきます。これを食べるとボーナス得点がもらえ、これは表示されます。フルーツは1つ10点、パワーエサは1つ50点です。10,000点を越えると、ワッマンが1つ増えます。

ゲームのスタートは9000番地からです。最初に得点とキートンが表され、[CR]キーを押すとゲームが始まります。ハイ・スコアをゼロにするときは、A98D番地とA98E番地を00にしてください。

2 プログラムの入力方法

プログラムはオール・マシン語です。モニタのMコマンドを使って入力してください。かなり長いプログラムなのでミスなに打ち込むのは難しいと思います。必ずチェック・サムを確認しておいてください。チェック・サムのプログラムは81年10月号のスペース・インベーダーにのっている

ものか使えます。

マシン語は9000・A98D番地までです。テープへのSAVEは\$コマンドを使います。S-ADR、\$に9000、E-ADR、\$にA98F、J-ADR、\$に9000としてテープを作れば、LOADしたときにはオート・スタートになります。

3 プログラムの解説

プログラムはオール・マシン語で、各処理はサブルーチン化し、図1のジェネラル・フローチャートに示すように順に呼び出していきます。各サブルーチンの頭でタイミングをとっていて、これによってスピードに差がつくようになっています。各サブルーチンの主なものについて図2～図7のフローチャートに示します。パックマンとモンスターの移動では、書ききれないため一部を省略してあります。メイン・ループ内、タイトル表示時、ゲーム・オーバー時に[BREAK]キーを押したかの判定をつけてあります。キーを押すとモニタに戻れます。

このプログラムでは、右側のスコアなどの文字や数字それにフルーツを食べたときのボーナス点などは、キャラクター画面に書き込んでいますが、その他の部分は、すべてグラフィック画面を使っています。しかし、グラフィック画面上で各種判定を行なうのは大変なことなので、ワーク・エリア内に40×25のワーク画面を作り、ここでドットとパワーエサ、壁などの判定を行なっています。モンスター、パックマン、フルーツはグラフィック画面上だけにあり、パックマンがモンスターを食べるなどの判定は、すべてアドレスを比較することによって行なっています。そのためパックマンはモンスターに半分かじられても逃げられます。もちろんその逆もあります。またモンスターの移動によりワーク画面を乱されることがないので、ドットがいつのまにか行方不明になるということはなく、ドットが0になれば終了します。

図1 ジェネラル・フローチャート

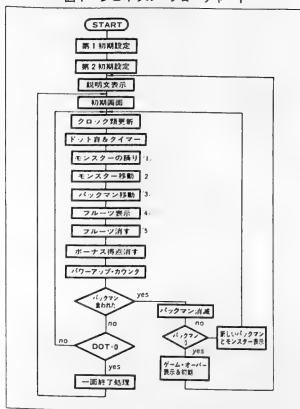
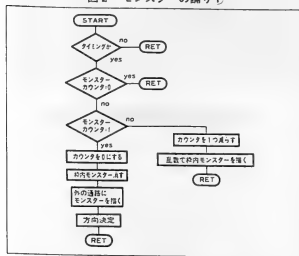


図2 モンスターの踊り①

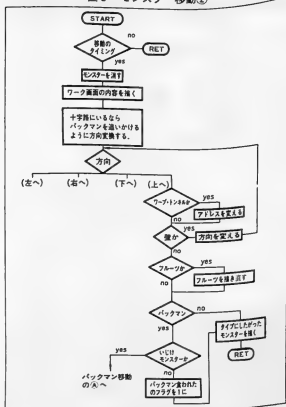


モンスターは水平方向のときは目玉が左右に動き、垂直方向のときは上下に動くようになっています。

モンスターの動きは乱数で決まるのですが、このときの条件が、それぞれのモンスターによって違うようにしてありますので、しつこく追いかけて来るのやフラフラしているのがあります。

パワーエサを食べるとモンスターは黒く(中抜け)になり逃げまわります。これが元に戻るときは点滅するので、このときに既に食われて中央の部屋に戻ったものまで一緒に点滅してしまいますが、気にしないでください。また、点滅はいじけモンスターと強力モンスターを交互に描いているため、このとき食いつく50%の確率で強力モンスターをかじるのになってバックマンはやられてしまいます。よって点滅し始めたところまで逆寄らない方が得です。モンスター

図3 モンスター移動②



スターのいじけている時間は局面が増すにつれて短くなります。

モンスターの移動スピードは最初のうちは遅いのですが、4面、8面、16面になったとき、1段階ずつスピード・アップしますから、だんだん難しくなります。これはワーク・エリアでなくプログラムの中を書き替えているので注意してください。

ドットの数が70個のときと140個のときにフルーツが出現します。フルーツは本物と同じようにしました。局面とフルーツの得点の関係を表1に示します。2面からはバナナでボーナス得点10,000点という意見もありますが、そうでないという意見もあり、ここは省略してしまいました。フルーツは一定時間たつと消えてしまいます。ベルとか鍵とかが出て来るとのにフルーツと呼ぶのはちょっと変な気がします。

ワーク・エリアを表2に、データ・エリアを表3に示します。キャラクタのデータやワーク画面初期用のデータがあるため、かなりRAMをくってしまいました。

4 ゲームの変更点

- ① 全体のスピードは、9 B 3 E 番地の値を変えてください。
小さくすると速くなります。
- ② バックマンのスピードは、9053番地の値を変えてください。
最初のうちはモンスターよりかなり速くしてあります。
- ③ モンスターのスピードは、局面が進むと速くなります。
その初期設定の値は、強力モンスターでは9 A 37番地、
いけばモンスターでは9 A 3 C 番地を変えてください。

A9AD	ドット=0になったかのフラグ
A9AF, A9B0	モンスターポイントの初期値および得点
A9B1	パワーUP期間の初期値
A9B3	パワーUPカウンターの値
A9B5	バックマンやられたときのフラグ
A9B6	フルーツが出現しているかのフラグ
A9B7, A9B8	フルーツの得点
A9B9	フルーツ得点を表示している期間
A9BA	フルーツ得点を表示しているかのフラグ
A9BB~A9DE	モンスターの待機カウンタ、Y座標、X座標、タイプ、方向、予備4バイトの計9バイトを4区画分
A9DF	フルーツを出現させるかのフラグ
A9E0	フルーツの出現時間
A9E1	モンスターのタイプ
A9E2	方向変換のための乱数値
A9E4, A9E5	バックマン消滅音のスタック
A9E7~ADD0	40×25の仮想画面

表3 データ・エリア

アドレス	内容
9F27	バックマン追加音
9F40	パワーエサ食べたときの音
9F4E	いじけモンスターを食ったときの音
9F56	フルーツを食べたときの音
9F75	一面終了時の音
9FF8	ゲーム・スタート時の音
A03F	ゲーム・オーバー時の音
A059	各種の壁のデータ
A0E0	パワーエサのデータ
A0E9	上下移動時のモンスターのデータ
A12D	左右移動時のモンスターのデータ
A171	いじけモンスターのデータ
A1B5	バックマンのデータ
A25F	消滅するバックマンのデータ
A2E9	各種フルーツのデータ
A3EC	初期画面のデータ
A725	グラフィック画面に壁を描くためのデータ(アドレス)
A7F4	画面に表示する文字のデータ
A822	説明文の文字データ
A946	ゲーム・オーバーの文字データ

筆者のハイ・スコアは4万点弱でした。友人の村井君の出した49,440点というのは驚異的だと思うのですが、皆さん挑戦してみてください。

□参考にしたもの

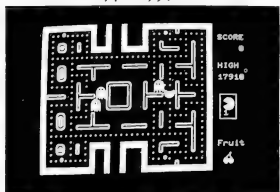
1)ゲーム「バックマン」

2)TOMグループ:「バックマン」, I/O, '81年4月号

ゲーム開始



アノバックマン...



バックマン プログラム・リスト

9000 7E 07 CD 50 0E CD 9F A9 176	9130 D5 E5 11 4F A1 18 0E C5 1A6	9260 CD F4 93 CD 7C 93 22 9F 1F1
9008 CD 0E 9A CD A1 9A CD A3 18D	9138 D5 E5 11 71 A1 18 0A C5 1C0	9268 A9 FD E9 00 00 00 CD C5 121
9010 99 CD B5 9C CD 4A 90 CD 1FB	9140 D5 E5 11 93 A1 18 CD A9 91 01	9270 91 7C FE 0E 20 0A 7D FE 1B6
9018 77 97 CD 4F 94 CD D9 94 1B8	9148 E1 D1 C1 C9 C5 D5 E5 11 1CC	9278 01 20 05 2E 18 22 9F A9 08E
9020 CD F2 91 CD 8F 9E CD DD 124	9150 85 A1 18 F1 C5 D5 E5 11 1EF	9280 FD 21 64 91 2D CD 52 9C 1FB
9028 9E CD 07 9F CD 64 97 3A 10F	9158 D7 A1 18 E9 C5 D5 E5 11 109	9288 FE 0A CA 4D 92 24 CD 52 1F4
9030 85 A9 FE 01 CA 97 9B 3A 10F	9160 F9 A1 18 E1 C5 D5 E5 11 123	9290 9C FE 0A CA 4D 92 CD FD 117
9038 AD A9 FE 01 CA 17 9B CD 19B	9168 1B A2 18 D9 C5 D5 E5 11 13E	9298 92 25 CD 52 9C CD FD 12C
9040 62 05 CA 00 00 18 CD 00 116	9170 2D A2 18 D1 C5 D5 E5 11 158	92A0 C3 60 92 00 00 00 CD C5 147
9048 00 00 21 9C A9 34 3A A2 16D	9178 5F A2 18 C9 C5 D5 E5 11 172	92A8 91 FD 21 54 91 25 CD 52 1D8
9050 A9 3C FE 04 20 01 AF 32 E19	9180 81 A2 18 C1 C5 D5 E5 11 18C	92B0 9C FE 0A CA 4D 92 CD 52 146
9058 A2 A9 3A A4 A9 FE 00 20 1F0	9188 C7 A2 18 89 C5 D5 E5 11 1CA	92B8 52 9C FE 0A CA 4D 92 CD 16C
9060 0F 3A AB A9 FE 01 28 0A 1C5	9190 A3 A2 18 0A C5 D5 E5 11 1F3	92C0 FD 92 2D CD 52 9C CD FD 141
9068 7E 08 1B 05 3E 0A 1B 01 1C4	9198 B5 A2 CD BB 91 E1 D1 C1 1E3	92C8 92 C5 60 92 00 00 00 CD 114
9070 3D 32 A4 A9 C9 00 00 00 1B5	91A0 A9 C9 00 00 00 00 CD 25 175	92D0 C5 91 FD 21 5C 91 24 CD 1A9
9078 C5 D5 E5 11 59 A0 1B 76 107	91AB 9C D1 0A 10 CD D6 9B E1 1A2	92D8 CD 52 9C FE 0A CA 4D 92 16C
9080 C5 D5 E5 11 62 A0 1B 2E 108	91AC 24 D5 CD 25 9C D1 0A 10 16E	92E0 2C CD 52 9C FE 0A CA 4D 106
9088 C5 D5 E5 11 6B A0 1B 26 109	91B8 C3 D6 9B 2C E5 CD CF 9B 7C	92E8 92 CD FD 92 2D CD 52 9C 1B6
9090 C5 D5 E5 11 74 A0 1B 1E 1DA	91C8 CD D5 91 E1 E5 24 CD D5 1B1	92F0 CD FD 92 2A 9F A9 24 CD 195
9098 C5 D5 E5 11 7D A0 1B 16 1DB	91D0 91 E1 D1 C1 C9 CD 25 9C 158	92F8 60 92 00 00 00 00 FE 1A 2B 12C
90A0 C5 D5 E5 11 8A A0 1B 0E 1DC	91D8 0E 10 AF 11 2B 00 CD CA 1BB	9300 34 FE 1E 0C E5 21 14 00 12A
90AB C5 D5 E5 11 8F A0 1B 0A 1DD	91E0 9B C5 D5 E5 1E 5B A9 A9 184	9308 22 AF A9 3E 01 32 AB A9 13C
90B0 C5 D5 E5 11 98 A0 CD CF 144	91E8 CD A4 91 E1 D1 C1 C9 00 13E	9310 3A 81 A9 32 BC A9 7E 0A 164
90B8 9B E1 D1 C1 C9 C5 D5 E5 156	91F0 00 00 3A A2 A9 FE 01 2B 16C	9318 DD 21 8B A9 DD 77 05 DD 196
90C0 11 A1 A0 1B F1 C5 D5 E5 16A	91F8 09 00 00 3A A2 A9 FE 01 2B 16C	9320 77 0C DD 77 15 DD 77 1E 15E
90C8 11 AA A0 1B E9 C5 D5 E5 1DB	9200 4C 91 2A 9F A9 0E 12 CD 13C	9328 D9 11 40 9F CD 3F 0F 1D 18D
90D0 11 B3 A0 1B E1 C5 D5 E5 1DC	9208 AD 9A CB 57 2B 18 CB 67 10B	9330 01 05 00 18 0A E5 01 01 108
90D8 11 BC A0 1B D9 C5 D5 E5 1DD	9210 CA A6 92 CB 77 CA CF 92 16F	9338 00 7E 01 32 AC A9 74 A6 196
90E0 11 C5 A0 1B D1 C5 D5 E5 1DE	9218 0E 11 CD AD 9A CB 57 D0 15	9340 49 3D 32 AC A9 FE 00 0A 196
90E8 11 CE A0 1B C9 C5 D5 E5 1DF	9220 C3 4E 92 00 00 00 CD C5 155	9348 07 7E 01 32 AD A9 1B 0D 1F3
90F0 11 D7 A0 1B C1 C5 D5 E5 1E0	9228 91 7C FE 0E 20 0A 7D FE 1B6	9350 FE 8C 2B 0A FE 46 20 05 11C
90F8 CD 25 9C 11 F0 00 19 76 1DE	9230 1B 20 05 2E 00 22 9F A9 1D5	9358 3E 01 32 DF A9 CD 69 93 1C2
9100 03 CD E2 9B 76 C5 E1 D1 138	9238 FD 21 6C 91 2C 2C CD 52 92	9360 E1 E5 0E 2E 9C 3A 00 E1 174
9108 CD C9 C5 D5 E5 11 E0 A0 9A	9240 9C FE 0A 2B 08 CD 52 117	9368 C9 2A 94 A9 09 22 94 A9 196
9110 C9 CF 9B E1 D1 C1 C9 C5 138	9248 9C FE 0A 20 05 2A 9F A9 1B8	9370 11 C1 CD CD FC 9A C5 6C 134
9118 D5 E5 11 E9 A0 1B 26 C5 157	9250 FD E9 CD 0D 25 CD 52 1B8	9378 9B 00 00 00 00 DD 21 B8 A9 194
9120 D5 E5 11 E9 A1 1B 1E C5 172	9258 9C CD FD 92 2A 9F A9 2C 176	9380 0E 0A DD 7E 01 8D 20 06 14D
9128 D5 E5 11 2D A1 1B 16 C5 18C		9388 DD 7E 02 BC 2B 0B 11 09 163

9C60 18 02 06 03 CD 78 90 24 11C
9C68 10 FA C9 06 05 18 02 06 1E
9C70 19 CD C9 06 20 1A FA C9 1E
9C78 06 03 18 02 06 02 CD 80 17B
9C80 90 24 10 FA C9 CD BA 9A 1A8
9C88 CD CE 9A CD C6 9A 11 EC 15F
9C90 A3 21 E7 A7 0E 19 06 21 1A2
9C98 E3 CD AB 7B CE 02 9B 12 23
9CA0 20 F3 21 01 00 CD 5E 1A0
9CA8 9C 21 01 01 CD 5E 9C 21 1B8
9CB0 19 06 CD 21 05 9C 21 19 15 12F
9CB8 CD 5E 9C 21 05 9C 02 62 12E
9CC0 9C 21 05 11 CD 62 9C 21 1B1
9CC8 15 0A CD 62 9C 21 15 11 13F
9CD0 CD 62 9C 21 01 0A CD 6B 12F
9CD8 9C 21 01 00 CD 6B 9C 21 10C
9CE0 01 10 CD 6B 9C 21 01 14 11B
9CE8 CD 6B 9C 21 15 0A CD 6B 14C
9CF0 9C 21 15 00 CD 6B 9C 21 1D4
9CF8 15 10 CD 6B 9C 21 15 14 14F
9D00 CD 6B 9C 21 01 00 CD 6B 14F 12
9D08 9C 21 01 20 CD 6C 9C 21 1D7
9D10 05 1B CD 7C 9C 21 08 0B 0C
9D18 CD 7C 9C 24 CD 7C 9C 21 10F
9D20 08 11 CD 7B 9C 21 08 1B 1E
9D28 CD 7C 9C 21 08 0E CD 7B 16C
9D30 9C 21 08 01 CD 7B 9C 21 1CD
9D38 08 0B CD 7C 9C 21 02 15 17B
9D40 CD 7C 9C 21 01 1B CD 7C 17F
9D48 9C 21 07 0E CD 7B 9C 21 10C
9D50 12 0B CD 7C 9C 21 12 0B 0C
9D58 CD 7C 9C 21 12 11 CD 7B 16E
9D60 9C 21 12 1B CD 7C 9C 21 1F0
9D68 15 1B CD 7C 9C 21 12 5F 1A
9D70 1A FE 00 2B 0B CD 9D 9E 150
9D78 CD 80 90 1B F3 11 80 A7 10B
9D80 1A FE 00 2B 0B CD 9D 9E 150
9D88 CD 90 90 1B F3 11 7B A7 12B
9D90 1A FE 00 2B 0B CD 9D 9E 150
9D98 CD 9B 90 1B F3 11 9A A7 14E
9DA0 1A FE 00 2B 0B CD 9D 9E 150
9DA8 CD 90 90 1B F3 11 AD A7 16D
9DB0 1A FE 00 2B 0B CD 9D 9E 150
9DB8 CD AB 90 1B F3 11 BC A7 18A
9DC0 1A FE 00 2B 0B CD 9D 9E 150
9DC8 CD 80 90 1B F3 11 8B A7 15C
9DD0 1A FE 00 2B 0B CD 9D 9E 150
9DD8 CD 80 90 1B F3 11 DB A7 185
9DE0 1A FE 00 2B 0B CD 9D 9E 150
9DE8 CD C5 90 1B F3 11 E5 A7 1CA
9DF0 1A FE 00 2B 0B CD 9D 9E 150
9DF8 CD CD 90 1B F3 21 07 07 16A
9E00 CD 05 90 21 0D 14 CD 05 11E
9E08 90 21 0D 14 CD 05 21 12 10F
9E10 08 1D CD 00 90 21 12 10 1AF
9E18 CD DD 90 21 12 0A CD 05 129
9E20 90 21 0B 0A CD 05 9D 00 CD 1DA
9E28 A3 9E 21 02 0A CD 0A 91 10D
9E30 21 17 04 CD 0A 91 21 02 0C7
9E38 1B CD 0A 91 21 17 18 CD 1A8
9E40 0A 91 21 72 00 11 F4 A7 19A
9E48 CD A1 9B 21 3A D1 11 9B 141
9E50 A7 CD A1 9B 21 02 02 11 1B8
9E58 01 AB CD 9E 9B 21 42 03 130
9E60 11 1B AB CD A1 9B 21 0E 10C
9E68 23 CD 5A 91 21 17 23 CD 1FD
9E70 E1 91 2A 9A A9 11 C1 0D 7B
9E78 CD FC 9A 2A 8D A9 11 89 150
9E80 D1 CD FC 9A 2A 99 A9 0A 17A
9E88 30 72 7B 02 7E 0A 32 46 19F
9E90 A9 21 02 18 23 9F A9 11 18
9E98 5A 91 00 00 00 67 13 1A 179
9EA0 67 1C 99 21 02 01 0E 1A 192
9EA8 05 0E 1F CD 52 9C 7E 1A 1D7
9EB0 CC F5 90 24 10 F5 E1 2C 187
9EB8 00 20 ED C9 00 00 0A 1D 10
9EC0 DF A9 FE 01 C0 AF 32 DF 107
9EC8 A9 21 0C 12 CD E1 91 3E 1A5
9ED0 01 32 BA A9 7E 02 3E 1A4
9ED8 A9 C9 00 00 0A BA A9 10B
9EE0 FE 01 C0 3A 92 A9 E6 03 11E
9EE8 CD 3A 0E 0A 9E 00 2B 05 1AE
9EF0 3D 32 CD A9 C9 21 0C 12 100
9EF8 CD C5 91 AF 32 8A A9 C9 12C
9F00 00 00 00 0A BA A9 FE 01 19C
9F08 CD 3A BA A9 FE 00 2B 05 187
9F10 3D 32 BA A9 7E 02 3E 1A4
9F18 AF 77 23 77 23 77 1F4
9F20 32 BA A9 C9 00 00 00 43 1A1
9F28 30 44 45 46 47 41 42 2B 1F4
9F30 43 52 23 41 23 47 23 46 10C
9F38 23 44 23 43 52 00 00 12C
9F40 23 41 30 41 47 46 45 47 1EE
9F48 23 47 41 00 00 00 24 44 129

9F50 30 20 46 20 47 20 41 20 1B2
9F58 42 43 44 45 23 46 47 1E1
9F60 23 41 42 00 00 00 2B 45 123
9F68 30 2B 46 2B 47 2B 46 1AF
9F70 23 44 0D 00 00 2B 43 116
9F78 52 30 2B 43 34 52 30 2B 1D1
9F80 43 34 52 30 2B 43 34 52 1E9
9F88 52 30 2B 43 34 52 30 2B 1C7
9F90 30 2B 43 34 52 30 2B 1D4
9F98 30 2B 43 34 52 30 2B 1C6
9FA0 2B 43 34 52 30 41 33 47 1D6
9FAB 2B 43 34 52 30 2B 43 34 1C6
9FAC 52 30 41 35 52 33 2B 43 1EB
9FAD 34 52 30 2B 43 34 52 30 1DA
9FAE 2B 43 34 52 30 41 33 47 1D6
9FBB 43 34 52 30 2B 43 34 52 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 52 30 2B 43 34 52 30 2B 1D6
9FBE 43 34 52 30 2B 43 34 52 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB8 2B 43 34 52 30 2B 43 34 1C6
9FBA 2B 43 34 52 30 2B 43 34 1C6
9FBB 2B 43 34 52 30 2B 43 34 1C6
9FBC 2B 43 34 52 30 2B 43 34 1C6
9FBD 2B 43 34 52 30 2B 43 34 1C6
9FBE 2B 43 34 52 30 2B 43 34 1C6
9FB0 2B 43 34 52 30 2B 43 34 1C6
9FB

PUCK MAN

バックマン プログラム・リスト

A530	00	14	0A	00	14	0A	00	14	6	
A53B	14	14	14	14	14	0A	00	14	182	
A540	00	00	00	0A	0A	0A	0A	132		
A54B	00	00	0A	00	14	14	14	15A		
A550	0A	00	14	14	14	14	0A	16E		
A55B	00	00	00	00	14	0A	00	11E		
A560	00	00	00	0A	00	00	00	10A		
A56B	0A	00	00	0A	00	00	00	114		
A570	00	0A	00	00	00	14	0A	12B		
A57B	0A	0A	0A	0A	00	14	0A	150		
A580	0A	0A	00	00	0A	00	00	12B		
A58B	00	00	00	0A	0A	0A	0A	132		
A590	00	00	0A	0A	0A	00	14	13C		
A59B	0A	00	14	14	14	14	14	17B		
A5A0	00	14	00	00	00	0A	0A	132		
A5AB	0A	0A	00	00	0A	00	14	13C		
A5B0	14	14	14	0A	00	14	14	182		
A5BB	14	0A	00	14	00	00	00	13C		
A5C0	14	0A	00	14	00	00	0A	13C		
A5CB	0A	0A	0A	00	00	0A	00	132		
A5D0	14	00	00	14	0A	00	14	146		
A5DB	00	14	0A	0A	00	14	0A	150		
A5E0	00	14	14	14	14	0A	00	15A		
A5EB	00	00	00	00	00	00	00	100		
A5F0	00	14	0A	00	14	14	14	16E		
A5FB	0A	00	14	0A	0A	00	14	150		
A600	00	00	14	00	00	0A	00	12B		
A60B	00	00	00	00	00	00	00	100		
A610	00	00	14	0A	00	14	00	132		
A61B	00	0A	00	14	0A	0A	00	146		
A620	0A	00	14	0A	0A	0A	150			
A62B	0A	0A	00	00	0A	0A	13C			
A630	0A	0A	00	14	0A	00	14	150		
A63B	0A	0A	0A	00	14	0A	0A	146		
A640	14	14	14	14	14	14	14	180		
A64B	14	14	14	14	14	14	14	180		
A650	14	14	14	14	14	14	14	180		
A65B	14	14	14	0A	00	14	0A	16E		
A660	00	14	00	00	14	00	00	12B		
A66B	14	00	00	00	00	00	00	114		
A670	00	00	00	00	00	14	00	114		
A67B	00	00	00	14	0A	00	14	13C		
A680	0A	00	14	0A	0A	14	150			
A68B	00	14	0A	0A	0A	00	13C			
A690	0A	0A	0A	0A	0A	00	14	150		
A69B	0A	0A	0A	00	14	0A	00	150		
A6A0	0A	0A	00	14	0A	0A	00	150		
A6AB	0A	00	14	0A	00	0A	00	132		
A6B0	00	0A	00	00	0A	00	14	12B		
A6BB	0A	00	14	14	14	14	0A	164		
A6C0	14	0A	0A	00	14	1E	14	182		
A6CB	14	14	14	14	0A	00	00	164		
A6D0	00	00	0A	00	00	00	0A	00	114	
A6DB	14	14	1E	14	00	00	14	182		
A6E0	14	14	0A	0A	00	00	00	13C		
A6EB	00	00	00	00	00	0A	00	10A		
A6F0	0A	00	00	0A	00	00	00	11E		
A6FB	00	00	00	00	00	0A	00	10A		
A700	00	00	00	0A	0A	0A	0A	132		
A70B	0A	0A	0A	0A	0A	0A	00	146		
A710	0A	0A	00	00	0A	00	00	114		
A71B	0A	0A	0A	0A	0A	0A	0A	150		
A720	0A	0A	0A	0A	1D	05	1D	171		
A72B	06	1D	07	03	09	04	09	14D		
A730	09	17	09	1D	09	07	0C	0D	16F	
A73B	0C	11	0C	14	0C	1A	0C	0D	17C	
A740	0D	11	0D	07	0E	0D	0E	11	16C	
A74B	0E	14	0E	1A	0E	03	11	04	170	
A750	11	0A	11	17	11	1D	11	1D	19F	
A75B	13	1D	14	1D	15	00	00	00	176	
A760	07	04	17	04	1D	04	17	0B	166	
A76B	07	0B	14	0B	04	0D	0A	0D	159	
A770	17	10	1D	10	07	15	1A	1B	1A2	
A77B	00	00	00	1A	02	07	05	0A	132	
A780	0A	17	0A	1D	0A	07	0F	14	17C	
A78B	0F	1A	0F	17	12	07	16	17	195	
A790	16	1D	16	00	00	00	1A	05	16B	
A79B	0D	0B	14	0B	04	0D	0A	0D	159	
A7A0	17	0D	1D	0D	12	14	12	12	193	
A7AB	1A	15	00	00	00	07	0B	10	14E	
A7B0	0B	1A	0B	07	12	10	12	1A	17F	
A7BB	12	00	00	00	03	04	03	0B	124	
A7C0	0D	0B	03	10	03	15	17	15	16F	
A7CB	00	00	00	04	04	04	0B	0A	11B	
A7D0	10	04	15	11	0B	00	00	00	145	
A7DB	04	05	0A	0A	04	12	04	16	147	
A7E0	11	0F	00	00	00	03	05	03	12B	
A7EB	0A	03	12	03	16	0D	0F	17	16B	
A7F0	05	00	00	00	53	43	4F	52	13C	
A7FB	45	00	00	4B	49	47	4B	00	165	
A800	00	97	9B	9B	95	00	9A	20	11C	
A80B	20	9A	00	9A	20	9A	00	9B	142	
A810	9A	20	20	9A	00	9B	9B	9B	142	
A81B	96	00	00	46	72	75	69	74	1A0	
A820	00	00	90	90	90	90	90	90	160	
A82B	90	90	90	90	90	90	90	90	180	
A830	90	90	90	00	2A	2A	2A	20	14E	
A83B	20	50	55	43	4B	4D	41	4E	2F	
A840	20	2A	2A	2A	2A	00	90	90	DE	
A84B	90	90	90	90	90	90	90	90	180	
A850	90	90	90	90	90	90	90	90	1F0	
A85B	00	00	23	23	23	20	53	43	11F	
A860	4F	52	45	20	41	44	56	41	122	
A86B	4E	43	45	20	54	41	42	4C	119	
A870	45	20	23	23	23	00	00	44	112	
A87B	4F	54	20	20	2D	2D	2D	2D	1A4	
A880	2D	2D	2D	2D	2D	2D	2D	31	15F	
A88B	50	20	70	6F	69	6E	74	73	1ED	
A890	00	00	50	4F	57	45	52	20	1AD	
A89B	46	45	45	44	20	2D	2D	2D	1BB	
A8A0	2D	2D	2D	35	30	20	70	6F	69	11A
A8AB	6E	74	73	00	00	46	52	55	142	
A8B0	49	54	20	2D	2D	2D	2D	2D	19E	
A8BB	2D	2D	2D	2D	2D	2D	2D	42	4F	4E
A8C0	55	53	20	70	6F	69	6E	74	1F2	
A8CB	73	00	00	31	30	30	30	30	164	
A8D0	20	70	6F	69	6E	74	73	20	1DD	
A8DB	A6	20	BA	84	D9	C4	20	50	141	
A8E0	55	43	4B	4D	41	4E	20	CC	1AB	
A8EB	84	0F	8D	A1	00	00	7C	20	13D	
A8F0	4B	45	59	20	46	75	6E	63	195	
A8FB	74	69	6F	6E	20	3E	20	97	1CF	
A900	9B	95	00	00	20	20	20	9A	12A	
A90B	3B	9A	00	97	9B	95	9B	9B	1CC	
A910	96	97	9B	95	00	9A	3A	9A	1C5	
A91B	20	2B	20	9A	3A	9A	00	9B	16D	
A920	9B	96	97	9B	95	9B	9B	9A	1C1	
A92B	00	20	20	20	9A	32	9A	00	1C6	
A930	20	20	20	9B	9B	96	00	00	129	
A93B	00	50	55	53	4B	20	43	52	1F5	
A940	20	6B	65	79	00	00	47	41	1F1	
A94B	4D	45	20	4F	56	45	52	00	1EE	
A950	00	00	00	00	20	50	55	43	10B	
A95B	4B	2D	2D	4D	41	4E	2D	2D	1E9	
A960	5A	2D	3B	30	42	29	20	20	19A	
A96B	56	45	52	20	21	2E	30	20	1BC	
A970	20	31	39	3B	31	2C	39	2C	184	
A97B	31	31	20	20	43	6F	70	79	13D	
A980	72	69	67	6B	74	20	62	79	119	
A98B	20	54	2E	53	20	00	00	00	115	



MZ-80B

DAN GAME

(PART1)



■ぱっくまん

ついに出来ました! I/O 読者が待ちに待ったDANのゲームができました! 雲はDANをどうするか? DANを殺すも生かすもあなた次第なのです。



GAMEの設定&説明

DANはI/O編集部から命令され、道に落ちている“□”を拾うことになった。しかし、昨日の痴漢がバレて2人の警官が捕まえようと追いかけてくる。また、ときどき女の子が横ぎるので、女に餌えるDANは痴漢をすればエネルギーになるのです。

さて、あなたもこの「DAN」になってみましょう!

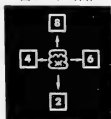
DANは図1の4つのキーで上下左右に動きます。警官2人が追いかけてくるので、捕まらないように逃げて“□”を取っていただく。また、ときどき女の子が横ぎるので、痴漢して(?)ください。BONUSになります。“□”を全部取ると1面クリアで、次の面が続きます。

3回捕まるとGAME OVERですが、5,000点を越えると1人増えます。

また、2面クリアするごとにコーヒープレイクがあります。そのDEMOは……ムフフ♡ まあ楽しみにしてやってみてください(しかし、中2がこんなものを作ってもいいのだろうか……)。

GAME OVERになってハイ・スコア5位以内に入っていると、NAMINGができます。さてお立ち会い! これはいままでのワンパターンとは違います。アプリケーションの

図1 キー操作



「music」の全パートをBGMに、[2]と[8]でアルファベットを決め、イニシャルでなくても3文字入力をするのです! (GALAGA式)。

あっそうそう、RUNするとイニシャルのため少し待ち、DEMOの途中に好きなキーを押し、HI-SCOREが表示され、もう一度押すとゲーム・スタートです。

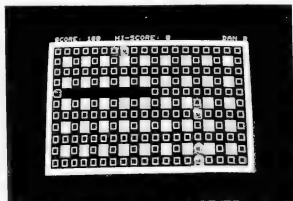
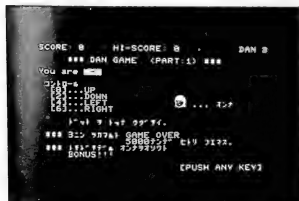


プログラム&Little Bug

フローチャートを図2、どこで何の処理かを表1に示します。

Little Bugとは小さな虫のことですが、これはときどき“□”が“コ”に化けたりすることです。これを取っても内容は無事だったということだけで得点になります。

また、途中での処理速度をあげるため、仮想画面を3枚持っています。スタート時のREADY!が表示している間にこれらを処理しているので、READY!状態が少し長いと思われるかもしれませんが、あらかじめご了承ください。




```

13 CURSOR20,24:PRINT"Made By Puck Man";
15 POKE$952,0:DEF KEY(5)=POKE$952,1667
20 LIMIT$CF7F:GOSUB9000:TEMP07
30 CURSOR0,0:PRINT" SCORE: 0 HI-SCORE:";HS(0);" ";HS$(0);TAB(35);"DAN 3
31 SCORE=0:LEVEL=0:DAN=3:UFRAG=1
35 GOSUB3000:GOSUB3500
40 GOSUB8500:P=164:PP=0
50 A=10:B=6:X=1:Y=1:V=1:W=1:FM=0:XX=0:XY=0:VX=0:VY=0
100 USR($E000):A0=19:B0=12:CURSORA0,B0:USR($E060)
110 USR($CF80,DA$(0))
111 X1=1:Y1=2:CURSORX1,Y1:USR($E060):USR($CF80,KK$)
112 V1=37:W1=22:CURSORV1,W1:USR($E060):USR($CF80,KK$)
113 IFSC=0THENMUSIC"E2DCEFGEDCEFGFEDFGAFEDFGAEDCEFGGEDCEFGAGFEDCEDCEDCR1
115 C$=" ":FORI=0TO5:K$=CHARACTER$(17+I,9):IFK$=CHR$(0)THENK$=" "
116 C$=C$+K$:NEXT:CURSOR17,9:PRINT"Ready!":IFP=164GOSUB8000:GOTO118
117 GOSUB8100:MUSIC"R7"
118 CURSOR17,9:PRINTC$
120 AA=0:BB=0:GETA$
130 IFA$="8" THENBB=-1
140 IFA$="2" THENBB=1
150 IFA$="4" THENAA=-1
160 IFA$="6" THENAA=1
170 IFAA+BB=0THENBB=-1
180 CURSORA0,B0:USR($CF80,CL$)
190 USR($E020):A0=19+AA:B0=12+BB:CURSORA0,B0:USR($E060)
195 FF=AA+ABS(AA)*2+BB+ABS(BB)*3
200 USR($CF80,DA$(FF))
210 FORI=1TO200:NEXT
230 CURSORA0,B0:USR($CF80,CL$):USR($E020):A0=19+AA*2:B0=12+BB*2:CURSORA0,B0:USR
($E060)
240 USR($CF80,DA$(FF))
290 GOTO500
300 IFDI(A,B)=3GOTO1500
301 IFDI(A,B)=4GOTO1200
305 A1=AA:B1=BB:AA=0:BB=0:GETA$
310 IFA$="8" THENBB=-1
320 IFA$="2" THENBB=1
330 IFA$="4" THENAA=-1
340 IFA$="6" THENAA=1
350 IFAA+BB=0THENAA=A1:BB=B1
355 F1=1
360 IFDI(A+AA,B+BB)THEN370
365 IFF1THENAA=A1:BB=B1:F1=0:GOTO360
366 GOSUB600:GOSUB800:GOTO300
370 CURSORA0,B0:USR($CF80,CL$):USR($E020):A0=A*2+AA-1:B0=B*2+BB:CURSORA0,B0:USR
($E060)
385 FF=AA+ABS(AA)*2+BB+ABS(BB)*3
390 USR($CF80,DA$(FF))
395 GOSUB600
400 GETA$
410 IFA$="8" THENB2=-1
420 IFA$="2" THENB2=1
430 IFA$="4" THENA2=-1
440 IFA$="6" THENA2=1
450 IFA2+B2=0THENA2=AA:B2=BB
470 CURSORA0,B0:USR($CF80,CL$):USR($E020):IFAA+A2+(BB+B2)*2THEN500
480 A0=A*2-1:B0=B*2:CURSORA0,B0:USR($E060):AA=0-AA:BB=0-BB
485 FF=AA+ABS(AA)*2+BB+ABS(BB)*3
490 USR($CF80,DA$(FF)):GOSUB800:GOTO300
500 A=A+AA:B=B+BB
505 IFDI(A,B)=3THEN1500
506 IFDI(A,B)=4THEN1200
510 A0=A*2-1:B0=B*2:CURSORA0,B0:USR($E060)
520 USR($CF80,DA$(FF))
530 IFDI(A,B)=2THEN550
540 GOSUB800:GOTO300
550 DI(A,B)=1:DJ(A,B)=1:USR($E080):CURSORA*2-1,B*2:USR($E050):P=P-1:IFP=PPGOTO2
000
560 SC=SC+10:CURSOR7,0:PRINTSC
565 IFUF$SC>5000THENUF=0:MUSIC"C0#CD#DEF#FG":DA=DA+1:CURSOR35,0:PRINT"DAN";DA
570 GOSUB800:GOTO300
600 IFRND(1)<(20-LE)*.01THENRE=2:RETURN
602 IFFMGOTO1000
605 X0=SGN(A-X):Y0=SGN(B-Y)
610 IFRND(1)<.1THENX0=SGN(A-X+RND(1)-.5)
620 IF(X0+XX)*X0=0THENY0=SGN(B-Y+.5):GOTO640

```

プログラム・リスト

```

630 IFDI(X+X0,Y) THEN XX=X0:XY=0:GOTO680
640 IFY0+XY=0GOTO660
645 IFY0GOTO650
646 X0=SGN(RND(1)-.5):Y0=SGN(RND(1)-.5)
650 IFDI(X,Y+Y0) THEN XY=Y0:XX=0:GOTO680
660 IFDI(X-X0,Y) THEN XX=-X0:XY=0:GOTO680
670 XX=0:XY=-Y0
680 CURSORX1,Y1:USR(¢F80,CL¢):USR(¢E020):X1=X*2-1+XX:Y1=Y*2+XY:CURSORX1,Y1:USR
(¢E060)
690 USR(¢F80,KK¢)
695 IFRND(1)<(20-LE)*.01 THEN RE=1:RETURN
700 V0=SGN(A-W):W0=SGN(B-W)
705 IFRND(1)<.1 THEN W0=SGN(B-W+RND(1)-.5)
710 IF(W0+VY)*W0=0 THEN V0=SGN(A-V+.5):GOTO730
720 IFDI(V,W+W0) THEN VX=0:VY=W0:GOTO770
730 IFV0+VX=0GOTO750
735 IFV0GOTO740
736 W0=SGN(RND(1)-.5):X0=SGN(RND(1)-.5)
740 IFDI(V+V0,W) THEN VX=W0:VY=0:GOTO770
750 IFDI(V,W-W0) THEN VX=0:VY=-W0:GOTO770
760 VY=0:VX=-V0
770 CURSORV1,W1:USR(¢F80,CL¢):USR(¢E020):V1=V*2-1+VX:W1=W*2+VY:CURSORV1,W1:USR
(¢E060)
780 USR(¢F80,KK¢)
790 RETURN
800 IFRE=2 THEN RE=0:GOTO840
802 IFFMGOTO1100
805 CURSORX1,Y1:USR(¢F80,CL¢):USR(¢E020)
810 DI(X,Y)=DJ(X,Y):X=X+XX:Y=Y+XY:IFDI(X,Y)<3 THEN DI(X,Y)=3:GOTO815
812 X=INT(RND(1)*6)*2+1:Y=INT(RND(1)*6)*2+1:IF((X=A)*(Y=B))+((X=W)*(Y=W))GOTO81
2
815 X1=X*2-1:Y1=Y*2:CURSORX1,Y1:USR(¢E060)
816 USR(¢F80,KK¢)
817 IFRE THEN RE=0:GOTO840
820 CURSORV1,W1:USR(¢F80,CL¢):USR(¢E020)
830 DI(V,W)=DJ(V,W):V=V+VX:W=W+VY:IFDI(V,W)<3 THEN DI(V,W)=3:GOTO835
832 V=INT(RND(1)*6)*2+1:W=INT(RND(1)*6)*2+1:IF((V=A)*(W=B))+((X=V)*(Y=W))GOTO83
2
835 V1=V*2-1:W1=W*2:CURSORV1,W1:USR(¢E060)
836 USR(¢F80,KK¢)
840 IF(INT(P/40)*40=P)*(1-FM)GOTO900
850 RETURN
900 FM=1:R=INT(RND(1)*4)
901 MUSIC"+C0+D+E+F+G
910 ONR/2+1GOSUB990,995
920 ONRGOTO940,950,960
930 H=0:GG=0:HH=1:RETURN
940 H=12:GG=0:HH=-1:RETURN
950 G=0:GG=1:HH=0:RETURN
960 G=20:GG=-1:HH=0:RETURN
990 G=INT(RND(1)*10)*2+1:RETURN
995 H=INT(RND(1)*6)*2+1:RETURN
1000 CURSORG1,H1:USR(¢F80,CL¢):USR(¢E020)
1010 CURSORX1,Y1:USR(¢E060)
1020 USR(¢F80,KK¢):DI(X,Y)=3
1030 CURSORV1,W1:USR(¢E060)
1040 USR(¢F80,KK¢):DI(V,W)=3
1060 G1=G*2-1+GG:H1=H*2+HH:CURSORG1,H1:USR(¢E060)
1070 USR(¢F80,MS¢)
1090 RETURN
1100 IFRN THEN RN=0:RETURN
1110 CURSORG1,H1:USR(¢F80,CL¢):USR(¢E020)
1115 DI(G,H)=DJ(G,H)
1120 G=GG+GG:H=HH+HH:IF(G=0)+(G=20)+(H=0)+(H=12) THEN FM=0:P=P-1:PP=PP-1:RETURN
1125 DI(G,H)=4
1130 G1=G*2-1:H1=H*2:CURSORG1,H1:USR(¢E060)
1140 USR(¢F80,MS¢)
1150 RETURN
1200 CURSORG1,H1:USR(¢F80,CL¢):PRINTSTR¢(LE+1):CURSORG1,H1+1:PRINT"00";
1220 DI(G,H)=DJ(G,H)
1230 FM=0:P=P-1:PP=PP-1
1240 SC=SC+LE*100+100:MUSIC"CODEFED
1242 CURSORG1,H1:USR(¢E020)
1245 IFUF$C$5000 THEN UF=0:MUSIC" C0#CD#DEF#FG":DA=DA+1:CURSOR35,0:PRINT"DAN";DA
1250 GOTO300
1500 A0=A*2-1:B0=B*2:CURSORA0,B0:USR(¢E060):USR(¢F80,KK¢)

```

```

1505 FORI=1TO3:MUSIC" +C0GBAGFEDCDEFG":NEXT
1510 DAN=DAN-1:CURSOR35,0:PRINT" DAN":DA
1512 DI(X,Y)=DJ(X,Y):DI(V,W)=DJ(V,W)
1513 GRAPHC:IFFM=1THENCURSORG1,H1:USR(0E020)
1514 CURSORX1,Y1:USR(0E020)
1515 CURSORV1,W1:USR(0E020)
1520 IFDAN>0GOTO50
1530 PRINTCHR$(6);:CURSOR0,10
1540 PRINT"
1550 PRINT"
1560 PRINT"
1570 PRINT"
1580 PRINT"
1590 IFSC>HS(0)THENCURSOR14,0:PRINT"HI-SCORE:";SC;" "
1595 IFSC>HS(4)GOTO1650
1600 FORI=1TO2000:NEXT:PRINTCHR$(6);
1610 FORI=1TO5:CURSOR10,I#4:PRINTJI$(I-1);TAB(24-LEN(STR$(HS(I-1)))):HS(I-1);
1620 PRINTTAB(30);HS$(I-1):NEXT
1630 FORI=1TO3000:NEXT:GOTO30
1650 FORI=1TO2000:NEXT:PRINTCHR$(6);
1655 FORI=4TO1STEP-1:HS(I+1)=HS(I):HS$(I+1)=HS$(I):IFHS(I-1)<SCTHENNEXT:HS(1)=H
B(0):HS$(1)=HS$(0):I=0
1656 WR=I:HS(WR)=0:HS$(WR)="0000"+CHR$(30)+"0000":HS(WR)=SC
1660 FORI=1TO5:CURSOR10,I#3:PRINTJI$(I-1);TAB(24-LEN(STR$(HS(I-1)))):HS(I-1);
1670 PRINTTAB(30);HS$(I-1):NEXT
1680 CURSOR14,20:PRINT"
1690 CURSOR14,21:PRINT"
1700 CURSOR14,22:PRINT"
1710 NA$="" :M=0:S=0:SF=0
1712 RESTORE9600
1715 MUSIC"R1"
1717 FORJ=1TO10:READM$:FORI=1TOLEN(M$):MM$=""
1720 S$=CHR$(S+65):IFS=26THENS$=""
1730 IFS=27THENS$=""
1740 CURSOR15+M$5,21:PRINTS$;
1750 GETA$:IFA$="" :8" THENS=S-1:IFS=-1THENS=27
1760 IFA$="" :2" THENS=S+1:IFS=28THENS=0
1770 IFA$<>" " THENSF=0:GOTO1980
1775 IFSFGOTO1980
1780 NA$=NA$+S$:M=M+1:S=0:SF=1:ONMGOTO1980,1980
1820 HS$(WR)=NA$
1830 PRINTCHR$(6);:FORI=1TO5:CURSOR10,I#3:PRINTJI$(I-1);TAB(24-LEN(STR$(HS(I-1)
)):HS(I-1);TAB(30);HS$(I-1):NEXTI
1835 CURSOR14,0:PRINT"HI-SCORE:";HS(0);" ";HS$(0)
1840 FORJ=1TO3000:NEXT:GOTO30
1980 IFMID$(M$,I,1)<"A" THENMM$=MM$+MID$(M$,I,1):I=I+1:GOTO1980
1990 MM$=MM$+MID$(M$,I,1):MUSICMM$:NEXT:RESTORE9600:GOTO1717
2000 MUSIC"CODEFGDEFGAEFGAB
2010 FORI=1TO5:MUSIC"FGAB+C":CONSOLE
2020 MUSIC"EFGBAB":CONSOLE:NEXT
2030 MUSIC"FGAB+CEFGABDEFGACDEFG"
2040 PRINTCHR$(6);:GRAPHIC
2050 FORI=1TO2000:NEXT
2060 LE=LE+1:IFINT(LE/2)*2<>LEGOTO40
2070 CONSOLE:FORI=0TO2:POSITION136,72+24*I:PATTERN-24,NU$(I):NEXT
2080 CURSOR14,22:PRINT"coffee break"
2090 FORI=1TO3:MUSIC"C1R9E1R9G1R9E1R9":NEXT
2100 CURSOR14,12:USR(0CF80,DA$(3))
2110 MUSIC"C1R9C1R9C1R9
2120 GRAPHIC:PRINTCHR$(6);
2130 CONSOLE
2400 GOTO40
3000 PRINTCHR$(6);:GRAPHIC
3008 FORI=0TO7
3010 LINE40-I,40-I,56,40-I,64+I,48,64+I,64,56,72+I,40-I,72+I,40-I,40-I
3020 LINE96-I,80,96-I,48,104,40-I,112,40-I,120+I,48,120+I,80
3025 LINE96,56+I,120,56+I
3030 LINE152-I,80,152-I,32:LINE152,40+I,176,64+I:LINE176+I,80,176+I,32
3035 GETA$:IFA$<>" " THENRETURN
3040 NEXT
3050 FORI=96TO143:LINE160,I,295,I:GETA$:IFA$<>" " THENRETURN
3055 NEXT
3060 BLINE192,104,168,104,168,135,192,135,192,127,184,127
3070 BLINE200,135,200,104,224,104,224,135:BLINE200,127,224,127
3080 BLINE232,135,232,104,255,104,255,135:BLINE244,104,244,128
3090 BLINE287,104,264,104,264,136,287,136:BLINE279,120,264,120

```

プログラム・リスト

```

3100 FORI=1T05:CONSOLE:MUSIC"+#B2":CONSOLE:MUSIC"+#B":GETA$:IFA$<>" THENRETUR
N
3105 NEXT
3110 FORI=1T05:GRAPH00:MUSIC"R":GRAPH01:MUSIC"R":GETA$:IFA$<>" THENRETURN
3115 NEXT:GRAPH00,C,01
3120 PRINTCHR$(6):PRINT"
3130 PRINT" You are LOUL":PRINT *** DAN GAME (PART:1) ***:PRINT
3140 PRINT" コントロール
3150 PRINT" [8]...UP
3160 PRINT" [2]...DOWN
3170 PRINT" [4]...LEFT
3180 PRINT" [6]...RIGHT:PRINT
3200 PRINT" トット ラ トット フク"サイ":PRINT
3220 PRINT" *** ジン マカルト GAME OVER":PRINTTAB(16);"5000テン" ヒトリ ファース.
3230 PRINT" *** トット マカルト オンマウスソフト
3240 PRINT" BONUS!!!
3250 PRINT:PRINT" [PUSH ANY KEY]
3260 CURSOR24,8:USR$(CFB0,MS$):CURSOR27,9:PRINT"... オン
3270 FORI=1T0200:GETA$:IFA$<>" THENRETURN
3280 NEXT
3290 CURSOR24,8:USR$(CFB0,KK$):CURSOR27,9:PRINT"... タイカン
3300 FORI=1T0200:GETA$:IFA$<>" THENRETURN
3310 NEXT
3315 FORJ=0T04
3320 CURSOR24,8:USR$(CFB0,DA$(J)):CURSOR27,9:PRINT"... LOUL "
3330 FORI=1T0200:GETA$:IFA$<>" THENRETURN
3340 NEXT:I,J
3350 GRAPH0:PRINTCHR$(6);
3360 FORI=1T05:CURSOR10,I#4:PRINTJI$(I-1);TAB(24-LEN(STR$(HS(I-1)))):HS(I-1);
3370 PRINTTAB(30);HS(I-1):NEXT
3380 FORI=1T01000:GETA$:IFA$<>" THENRETURN
3390 NEXT
3400 GOTO3000
3500 GRAPH0:PRINTCHR$(6);
3510 FORI=1T05:CURSOR10,I#4:PRINTJI$(I-1);TAB(24-LEN(STR$(HS(I-1)))):HS(I-1);
3520 PRINTTAB(30);HS(I-1):NEXT
3530 GETA$:IFA$<>" THEN3530
3540 GETA$:IFA$<>" THENRETURN
3550 GOTO3540
7998 END
7999 REM *** ダーイチ ショキ マッティ ***
8000 FORI=1T019:FORJ=1T011:DI(I,J)=2:NEXTJ,I
8010 FORI=2T018STEP2:FORJ=2T010STEP2:DI(I,J)=0:NEXTJ,I
8030 FORI=0T020:FORJ=0T012:DJ(I,J)=DI(I,J):NEXTJ,I
8040 RETURN
8100 FORI=0T020:FORJ=0T012:DI(I,J)=DJ(I,J):NEXTJ,I
8110 RETURN
8499 REM *** ショキ カ マン ***
8500 PRINTCHR$(6);STRING$(CHR$(30),40);
8510 FORI=1T05
8520 FORJ=1T02:PRINTCHR$(30);SPACE$(38);CHR$(30);:NEXT
8530 FORJ=1T02:PRINTCHR$(30);" ";:FORK=1T09:PRINTCHR$(31);CHR$(31);" ";
8540 NEXT:PRINTCHR$(30);:NEXTJ,I
8550 FORI=1T02:PRINTCHR$(30);SPACE$(38);CHR$(30);:NEXT
8560 PRINTSTRING$(CHR$(30),39);CHR$(4);CHR$(8);CHR$(30);CHR$(5)
8570 FORI=2T022STEP4:FORJ=1T037STEP2:CURSORJ,I:PRINT" ";:CURSORJ,I+1:PRINT" "
;
8580 NEXTJ,I:FORI=4T020STEP4:FORJ=1T037STEP4
8590 CURSORJ,I:PRINT" ";:CURSORJ,I+1:PRINT" ";:NEXTJ,I
8600 RETURN
9000 RESTORE
9005 FORI=0T016#9-1:READY:POKE57344+I,Y:NEXT
9010 RESTORE9900:FORI=0T063:READY:POKE53120+I,Y:NEXT
9100 CL$=STRING$(CHR$(0),32)
9105 C0$=CHR$(0)+CHR$(0):C2$=CHR$(2)+CHR$(2):C3$=CHR$(3)+CHR$(3)
9106 C4$=CHR$(4)+CHR$(4)
9110 DA$(3)=C0$+"$"+CHR$(4)+CHR$(2)+"+"+CHR$(12)+STRING$(CHR$(2),3)+CHR$(12)+C2
$+"$"+C0$
9120 DA$(3)=DA$(3)+C0$+CHR$(3)+CHR$(12)+CHR$(19)+"$ " $7770 "+CHR$(16)+CHR$(15)+C0
$
9130 DA$(1)=C0$+"70"+CHR$(8)+"$"+CHR$(2)+C3$+CHR$(2)+CHR$(4)+CHR$(8)+"$"+C0$
9140 DA$(1)=DA$(1)+C0$+CHR$(31)+" 001000000?" +C0$
9150 DA$(2)="117 "+"CHR$(16)+"$"+CHR$(8)+"$"+STRING$(CHR$(4),5)+"$DHD"+CHR$(0)
9160 DA$(2)=DA$(2)+CHR$(1)+CHR$(3)+CHR$(4)+CHR$(8)+CHR$(16)+CHR$(26)+"$ " +C
HR$(34)+CHR$(18)+CHR$(13)+CHR$(0)
9170 DA$(4)=CHR$(0)+"DHD"+STRING$(CHR$(4),5)+"$X"+CHR$(8)+CHR$(16)+" 711"

```


編集後記

■MZ-80Bは64K オールRAM 構成というクリーン・コンピュータで高いパフォーマンスと柔軟性を持っています。そのため、ソフトでそのパフォーマンスを引き出せるという意味でマニア好みのマイコンともいえるでしょう。ここではコンパイラ、インタープリタ、アセンブラ、デバッガなどのソフト開発ツール、プログラムレス言語などのビジネス・プログラム、各種アプリケーション、ゲーム・プログラムなど多数収録しま

した、これらのソフトが何らかのお役に立てば幸いです。

■本書は月刊誌I/Oに発表された記事が中心ですが、モニタ・サブルーチンの活用法といったソフト開発に必要な情報が掲載されています。ユーザーには参考になる点が多いと思うので、ぜひご一読を。

■なお、現在、毎月多くのMZBに関する記事が掲載されており、『MZ-80B活用研究』も続編を予定しています。

本書やI/Oをご覧になってMZBに関する新しいアイデアが出たら、ぜひI/Oに投稿してください。よろしくお願いいたします。

I/O別冊

MZ-80B活用研究

昭和57年10月7日 初版発行

定価2500円

©1982

編集集 I/O編集部

発行人 星 正明

発行所 株式会社 工学社

〒151 東京都渋谷区代々木1-37-1 ぜんらくビル5F

☎03375-5784代 [営業・広告]

☎03320-1218代 [編集]

振替 東京5-22510



MZ-80B活用研究

定価2500円 雑誌61473-49